

Vadim Ermolayev
Heinrich C. Mayr
Mykola Nikitchenko
Aleksander Spivakovskiy
Grygoriy Zholtkevych
Mikhail Zavileysky
Hennadiy Kravtsov
Vitaliy Kobets
Vladimir Peschanenko
(Eds.)



ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer

Proceedings of the 9th International Conference,
ICTERI 2013

Kherson, Ukraine
June, 2013

Ermolayev, V., Mayr, H. C., Nikitchenko, M., Spivakovskiy, A., Zholtkevych, G., Zavileysky, M., Kravtsov, H., Kobets, V. and Peschanenko, V. (Eds.): ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer. Proc. 9th Int. Conf. ICTERI 2013, Kherson, Ukraine, June 19-22, 2013, CEUR-WS.org, online

This volume constitutes the refereed proceedings of the 9th International Conference on ICT in Education, Research, and Industrial Applications, held in Kherson, Ukraine, in June 2013.

The 49 papers were carefully reviewed and selected from 124 submissions. The volume opens with the contributions of the invited speakers. Further, the part of the volume containing the papers of the main ICTERI conference is structured in four topical parts: ICT infrastructures, Integration and Interoperability; Machine Intelligence, Knowledge Engineering and Management for ICT; Model-based software system development; and Methodological and Didactical Aspects of Teaching ICT and Using ICT in Education. This part of the volume is concluded by the two papers describing the tutorials presented at the conference. The final part of the volume comprises the selected contributions of the three workshops co-located with ICTERI 2013, namely: the 1st International Workshop on Methods and Resources of Distance Learning (MRDL 2013); the 2nd International Workshop on Information Technologies in Economic Research (ITER 2013); and the 2nd International Workshop on Algebraic, Logical, and Algorithmic Methods of System Modeling, Specification and Verification (SMSV 2013).

Copyright © 2013 for the individual papers by the papers' authors.
Copying permitted only for private and academic purposes. This volume is published and copyrighted by its editors.

Preface

It is our pleasure to present you the proceedings of ICTERI 2013, the ninth edition of the International Conference on Information and Communication Technologies in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer, held at Kherson, Ukraine on June 19-22, 2013. ICTERI is concerned with interrelated topics from information and communication technology (ICT) infrastructures to teaching these technologies or using those in education or industry. Those aspects of ICT research, development, technology transfer, and use in real world cases are vibrant for both the academic and industrial communities.

The conference scope was outlined as a constellation of the following themes:

- ICT infrastructures, integration and interoperability
- Machine Intelligence, knowledge engineering, and knowledge management for ICT
- Cooperation between academia and industry in ICT
- Model-based software system development
- Methodological and didactical aspects of teaching ICT and using ICT in education

A visit to Google Analytics proves the broad and continuously increasing interest in the ICTERI themes. Indeed, between November 15, 2012 and May 15, 2013 we have received circa 4 400 visits to the conference web site, <http://icteri.org/>, from 110 countries (568 cities). These numbers are 1.5 – 2 times higher than those observed in the similar period for ICTERI 2012.

ICTERI 2013 continues the tradition of hosting co-located focused events under its umbrella. In the complement to the main conference this year, the program offered the three co-located workshops, two tutorials, and IT talks panel. The main conference program has been composed of the top-rated submissions evenly covering all the themes of ICTERI scope.

The workshops formed the corolla around the main ICTERI conference by focusing on particular sub-fields relevant to the conference theme. In particular:

- The 1st International Workshop on Methods and Resources of Distance Learning (MRDL 2013) dealt mainly with the methodological and didactical aspects of teaching ICT and using ICT in education
- The scope of the 2nd International Workshop on Information Technologies in Economic Research (ITER 2013) was more within the topic of cooperation between academia and industry

II

- 2nd International Workshop on Algebraic, Logical, and Algorithmic Methods of System Modeling, Specification and Verification (SMSV 2013) focused on model-based software system development

The IT Talks panel was the venue for the invited industrial speakers who wish to present their cutting edge ICT achievements.

This year we were also accepted the two focused short tutorials to the program: on ontology alignment and the industrial applications of this technology; and on the time model and Clock Constraint Specification Language for the UML profile used in modeling and analysis of real-time and embedded systems.

Overall ICTERI attracted a substantial number of submissions – a total of 124 comprising the main conference and workshops. Out of the 60 paper submissions to the main conference we have accepted 22 high quality and most interesting papers to be presented at the conference and published in our proceedings. The acceptance rate was therefore 36.7 percent. Our three workshops received overall 64 submissions, from which 27 were accepted by their organizers and included in the second part of this volume. Those selected publications are preceded by the contributions of our invited speakers. The talk by our keynote speaker Wolf-Ekkehard Matzke expressed his industrial views on the knowledge-based bio-economy and the “Green Triple-Helix” of biotechnology, synthetic biology, and ICT. The invited talk by Gary L. Pratt was focused on a movement of higher education institutions to forming consortiums for creating a position of strength facing contemporary economic challenges. The invited talk by Alexander A. Letichevsky presented a general theory of interaction and cognitive architectures based on this theory.

The conference would not have been possible without the support of many people. First of all we would like to thank all the authors who submitted papers to ICTERI 2013 and thus demonstrated their interest in the research problems within our scope. We are also very grateful to the members of our Program Committee for providing timely and thorough reviews and also for been cooperative in doing additional review work. We would like also to thank the local organizers of the conference whose devotion and efficiency made this instance of ICTERI a very comfortable and effective scientific forum. Finally a special acknowledgement is given to the support by our editorial assistant Olga Tatarintseva who invested a considerable effort in checking and proofing the final versions of our papers.

June, 2013

Vadim Ermolayev
Heinrich C. Mayr
Mykola Nikitchenko
Aleksander Spivakovskiy
Grygoriy Zholtkevych
Mikhail Zavileysky
Hennadiy Kravtsov
Vitaliy Kobets
Vladimir Peschanenko

Organization

Organizers

Ministry of Education and Science of Ukraine
Kherson State University, Ukraine
Alpen-Adria-Universität Klagenfurt, Austria
Zaporizhzhya National University, Ukraine
Institute of Information Technology and Teaching Resources, Ukraine
V. N. Karazin Kharkiv National University, Ukraine
Taras Shevchenko National University of Kyiv, Ukraine
DataArt Solutions Inc.

General Chair

Aleksander Spivakovsky, Kherson State University, Ukraine

Steering Committee

Vadim Ermolayev, Zaporizhzhya National University, Ukraine
Heinrich C. Mayr, Alpen-Adria-Universität Klagenfurt, Austria
Natalia Morse, National University of Life and Environmental Sciences, Ukraine
Mykola Nikitchenko, Taras Shevchenko National University of Kyiv, Ukraine
Aleksander Spivakovsky, Kherson State University, Ukraine
Mikhail Zavileysky, DataArt, Russian Federation
Grygoriy Zholtkevych, V.N.Karazin Kharkiv National University, Ukraine

Program Co-chairs

Vadim Ermolayev, Zaporizhzhya National University, Ukraine
Heinrich C. Mayr, Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria

Workshops Chair

Mykola Nikitchenko, *Taras Shevchenko National University of Kyiv, Ukraine*

Tutorials Chair

Grygoriy Zholtkevych, *V.N.Karazin Kharkiv National University, Ukraine*

IT Talks Co-chairs

Aleksander Spivakovsky, *Kherson State University, Ukraine*

Mikhail Zavileysky, *DataArt, Russian Federation*

Program Committee

Rajendra Akerkar, *Western Norway Research Institute, Norway*

Eugene Alferov, *Kherson State University, Ukraine*

Costin Badica, *University of Craiova, Romania*

Tobias Buerger, *PAYBACK, Germany*

Andrey Bulat, *Kherson State University, Ukraine*

David Camacho, *Universidad Autonoma de Madrid, Spain*

Michael Cochez, *University of Jyväskylä, Finland*

Maxim Davidovsky, *Zaporizhzhya National University, Ukraine*

Anatoliy Doroshenko, *National Technical University of Ukraine "Kyiv Polytechnic Institute", Ukraine*

Vadim Ermolayev, *Zaporizhzhya National University, Ukraine*

David Esteban, *Techforce, Spain*

Lyudmila Gavrilova, *Slovyansk State Pedagogical University, Ukraine*

Vladimir Gorodetsky, *St. Petersburg Institute for Informatics and Automation of The Russian Academy of Science, Russian Federation*

Marko Grobelnik, *Jozef Stefan Institute, Slovenia*

Brian Hainey, *Glasgow Caledonian University, United Kingdom*

Sungkook Han, *Wonkwang University, South Korea*

Mitja Jermol, *Jozef Stefan Institute, Slovenia*

Jason Jung, *Yeungnam University, South Korea*

Natalya Keberle, *Zaporizhzhya National University, Ukraine*

Nick Kings, *Connected Shopping, United Kingdom*

Christian Kop, *Alpen-Adria-Universität Klagenfurt, Austria*

Hennadiy Kravtsov, *Kherson State University, Ukraine*

Nataliya Kushnir, *Kherson State University, Ukraine*

Frédéric Mallet, *Université de Nice-Sophia Antipolis, France*

Mihhail Matskin, *Royal Institute of Technology, Sweden*

Heinrich C. Mayr, *Alpen-Adria-Universität Klagenfurt, Austria*

Mykola Nikitchenko, *Taras Shevchenko National University of Kyiv, Ukraine*

Andriy Nikolov, *Knowledge Media Institute, The Open University, United Kingdom*

Inna Novalija, *Jozef Stefan Institute, Slovenia*

Tomás Pariente Lobo, *ATOS Origin, Spain*

Vladimir Peschanenko, *Kherson State University, Ukraine*

Carlos Ruiz, *playence, Spain*

Abdel-Badeeh Salem, *Ain Shams University, Cairo, Egypt*

Wolfgang Schreiner, *Research Institute for Symbolic Computation (RISC),*

Johannes Kepler University, Austria

Vladimir A. Shekhovtsov, *Alpen-Adria-Universität Klagenfurt, Austria*
Mikhail Simonov, *Istituto Superiore Mario Boella, Italy*
Marcus Spies, *Ludwig-Maximilians-Universität München, Germany*
Aleksander Spivakovsky, *Kherson State University, Ukraine*
Martin Strecker, *IRIT, Université Paul Sabatier, France*
Olga Tatarintseva, *Zaporizhzhya National University, Ukraine*
Vagan Terziyan, *University of Jyväskylä, Finland*
Nikolay Tkachuk, *National Technical University "Kharkiv Polytechnic Institute", Ukraine*
Leo Van Moergestel, *Utrecht University of Applied Sciences, Netherlands*
Maryna Vladymyrova, *V. N. Karazin Kharkov National University, Ukraine*
Paul Warren, *Knowledge Media Institute, the Open University, United Kingdom*
Iryna Zaretska, *V. N. Karazin Kharkov National University, Ukraine*
Mikhail Zavileysky, *DataArt Solutions Inc., Russian Federation*
Grygoriy Zholtkevych, *V. N. Karazin Kharkov National University, Ukraine*

Additional Reviewers

Fahdi Al Machot, *Alpen-Adria-Universität Klagenfurt, Austria*
Antonio Gonzalez-Pardo, *Universidad Autonoma de Madrid, Spain*
Alexey Vekschin, *National Technical University "Kharkiv Polytechnic Institute", Ukraine*

Sponsors

Organizations and Companies



DataArt (<http://dataart.com/>) develops custom applications, helping clients optimize time-to-market and save costs



Kherson State University (<http://www.ksu.ks.ua/>) is a multidisciplinary scientific, educational, and cultural center in the south of Ukraine



Zaporizhzhya National University (<http://www.znu.edu.ua/>) is a renowned educational and research center of Ukraine that offers a classically balanced diversity of high quality academic curricula and many opportunities to build your scientific carrier

Individuals



Aleksandr Spivakovsky is the chair of the Department of Informatics and the first vice-rector of Kherson State University



Dmitriy Shchedrolov is the head of DataArt's R&D Center at Kherson

Table of Contents

Preface	I
Organization.....	III
Sponsors.....	VI
Invited Contributions	1
The Knowledge-Based Bio-Economy and the “Green Triple-Helix” of Biotechnology, Synthetic Biology and ICT	2
A Movement of Higher Education Institutions to Consortiums of Institutions Banding Together to Create a Position of Strength.....	3
General Theory of Interaction and Cognitive Architectures	4
Part 1. Main ICTERI Conference	16
1.1 ICT Infrastructures, Integration and Interoperability	17
Wireframe Model for Simulating Quantum Information Processing Systems.....	18
Modeling, Algorithms and Implementation of the Microcontroller Control System for the Ion Beam Forming Process for Nanostructures Etching	30
Using Algebra-Algorithmic and Term Rewriting Tools for Developing Efficient Parallel Programs	38
1.2 Machine Intelligence, Knowledge Engineering and Management for ICT.....	47
An Intelligent Approach to Increase Efficiency of IT-Service Management Systems: University Case-Study	48
Refining an Ontology by Learning Stakeholder Votes from their Texts	64
Answering Conjunctive Queries over a Temporally-Ordered Finite Sequence of ABoxes sharing one TBox.....	79
An Adaptive Forecasting of Nonlinear Nonstationary Time Series under Short Learning Samples	91
Application of an Instance Migration Solution to Industrial Ontologies	99
Extracting Knowledge Tokens from Text Streams	108
1.3 Model-Based Software System Development.....	117

Use of Neural Networks for Monitoring Beam Spectrum of Industrial Electron Accelerators.....	118
Lazy Parallel Synchronous Composition of Infinite Transition Systems	130
Selecting Mathematical Software for Dependability Assessment of Computer Systems Described by Stiff Markov Chains	146
Asymptotical Information Bound of Consecutive Qubit Binary Testing.....	163
A Data Transfer Model of Computer-Aided Vehicle Traffic Coordination System for the Rail Transport in Ukraine	178
Quantitative Estimation of Competency as a Fuzzy Set	187
1.4 Methodological and Didactical Aspects of Teaching ICT and Using ICT in Education.....	194
New Approaches of Teaching ICT to Meet Educational Needs of Net Students Generation.....	195
Pedagogical Diagnostics with Use of Computer Technologies	209
The Use of Distributed Version Control Systems in Advanced Programming Courses.....	221
Comparative Analysis of Learning in Three-Subjective Didactic Model	236
Conception of Programs Factory for Representing and E-Learning Disciplines of Software Engineering	252
Public Information Environment of a Modern University	264
Designing Massive Open Online Courses.....	273
The Role of Informatization in the Change of Higher School Tasks: the Impact on the Professional Teacher Competences.....	281
1.5 ICTERI Tutorials	288
UML Profile for MARTE: Time Model and CCSL.....	289
Ontology Alignment and Applications in 90 Minutes	295
Part 2. ICTERI Workshops	307
2.1 2 nd International Workshop on Information Technologies in Economic Research (ITER 2013).....	308
Foreword.....	309
Binary Quasi Equidistant and Reflected Codes in Mixed Numeration Systems....	311
Mechanism Design for Foreign Producers of Unique Homogeneity Product.....	329
Features of National Welfare Innovative Potential Parametric Indication Information-Analytical Tools System in the Globalization Trends' Context	339
Matrix Analogues of the Diffie-Hellman Protocol	352
Are Securities Secure: Study of the Influence of the International Debt Securities on the Economic Growth.....	360
How to Make High-tech Industry Highly Developed? Effective Model of National R&D Investment Policy	366

Econometric Analysis on the Site “Lesson Pulse”	374
Decision Supporting Procedure for Strategic Planning: DEA Implementation for Regional Economy Efficiency Estimation	385
Applying of Fuzzy Logic Modeling for the Assessment of ERP Projects Efficiency	393
Mathematical Model of Banking Firm as Tool for Analysis, Management and Learning	401
2.2 1 st International Workshop on Methods and Resources of Distance Learning (MRDL 2013)	409
Foreword.....	410
What Should be E-Learning Course for Smart Education	411
TIO – a Software Toolset for Mobile Learning in MINT Disciplines	424
Holistic Approach to Training of ICT Skilled Educational Personnel.....	436
2.3 2 nd International Workshop on Algebraic, Logical, and Algorithmic Methods of System Modeling, Specification and Verification (SMSV 2013)	446
Foreword.....	447
An Abstract Block Formalism for Engineering Systems	448
Multilevel Environments in Insertion Modeling System	464
Clocks Model for Specification and Analysis of Timing in Real-Time Embedded Systems	475
Specializations and Symbolic Modeling	490
On a Dynamic Logic for Graph Rewriting	506
Logical Foundations for Reasoning about Transformations of Knowledge Bases	521
Program Algebras with Monotone Floyd-Hoare Composition	533
A Formal Model of Resource Sharing Conicts in Multithreaded Java	550
Implementation of Propagation-Based Constraint Solver in IMS.....	565
UniTESK: Component Model Based Testing.....	573
Protoautomata as Models of Systems with Data Accumulation	582
Models of Class Specification Intersection of Object-Oriented Programming.....	590
Author Index	595

Invited Contributions

The Knowledge-Based Bio-Economy and the “Green Triple-Helix” of Biotechnology, Synthetic Biology and ICT

Wolf-Ekkehard Matzke

MINRES Technologies GmbH, Neubiberg, Germany

wolf@minres.com

Abstract. Over the last decades economies around the globe have transformed into a knowledge-based economy (KBE). Information and Communication Technology (ICT) has served as the principal enabling technology for this transformation. Now biology becomes another major pillar – producing a knowledge-based bio-economy (KBBE). The challenges faced by biotechnology push the requirements for ICT in many ways to the extreme and far beyond its basic utility function. In particular, it is valid for synthetic biology which aims to break ground on the rational design and construction of artificial biological systems with ICT as its backbone for bio-design automation (BDA). This could be best illustrated using a metaphor of a “green triple-helix”, where “green” stands for environmental consciousness and “triple-helix” visualizes the interdependency of biotechnology, synthetic biology, and ICT as the helical strands. The talk will explore this inter-dependency in dynamics. High level ICT requirements will be identified and discussed along the dimensions of education, research and industry with the emphasis on synthetic biology and BDA. The guidelines for the architecture and implementation of an open BDA platform will be presented so that interested ICT researchers and practitioners will better understand the biology-specific ICT challenges of the KBBE.

Keywords. Knowledge-Based Bio-Economy, Biotechnology, Synthetic Biology, Bio-Design Automation

Key terms. ICTInfrastructure, Industry, Management, Research

A Movement of Higher Education Institutions to Consortiums of Institutions Banding Together to Create a Position of Strength

Gary L. Pratt

Eastern Washington University, 202 Huston Hall, Cheney, Washington 99004, USA

gpratt@ewu.edu

Abstract. Colleges and universities compete for students, faculty, and business, industry, and research partnerships with quality programs, strong faculty, research opportunities, affordable cost, and high student success factors. Yet, at the infrastructure level, most of these institutions provide many similar information technology services and support. On top of this, many of these institutions struggle to provide this quality infrastructure because of a variety of factors, including: shrinking budgets, minimal strategic planning, and a lack of institutional vision of information technology as a strategic asset. This presentation will showcase best practice examples of how higher education institutions can band together, to create strong consortium relationships that can help all partners in this relationship move forward as a strong force. Examples will include actual successes experience by the Kentucky Council on Postsecondary Education's Distance Learning Advisory Committee (DLAC), the Washington Legislative Technology Transformation Taskforce (TTT), and the Washington Higher Education Technology Consortium (WHETC). These successes range from statewide strategic planning efforts, to significant consortial purchasing contracts, to collaborative technology systems, services, and training opportunities. This presentation will show that institutions can be stronger working together than working individually.

Keywords. University consortium, best practice, competition, infrastructure, information technology, strategic asset, strategic planning, collaborative technology system

Key terms. Academia, Information Technology, Infrastructure, Cooperation, Management

General Theory of Interaction and Cognitive Architectures

Alexander Letichevsky

Glushkov Institute of Cybernetics, Academy of Sciences of Ukraine
40 Glushkova ave., 03187, Kyiv, Ukraine
`let@cyfra.net`

Abstract. The challenge of creating a real-life computational equivalent of the human mind is now attracting the attention of many scientific groups from different areas of cybernetics and Artificial Intelligence such as computational neuroscience, cognitive science, biologically inspired cognitive architectures etc. The paper presents a new cognitive architecture based on insertion modeling, one of the paradigms of a general theory of interaction, and a basis for multiagent system development. Insertion cognitive architecture is represented as a multilevel insertion machine which realizes itself as a high level insertion environment. It has a center to evaluate the success of its behavior which is a special type agent that can observe the interaction of a system with external environment. The main goal of a system is achieving maximum success repeated. As an agent this machine is inserted into its external environment and has the means to interact with it. The internal environment of intelligent cognitive agent creates and develops its own model and the model of external environment. If the external environment contains other agents, they can be modeled by internal environment which creates corresponding machines and interprets those machines using corresponding drivers, comparing the behaviors of models and external agents. Insertion architecture is now under development on the base of Insertion modeling system, developed in Glushkov Institute of Cybernetics.

Keywords. AgentBasedSystem, DistributedArtificialIntelligence, Reasoning, FormalMethod, Simulation

Key terms. AgentBasedSystem, DistributedArtificialIntelligence, Reasoning, FormalMethod, Simulation

1 Introduction

General theory of interaction is a theory of information interaction in complex distributed multi-agent systems. It has a long history. Contemporary part of this history can be considered as starting from neuro networks of McCulloch-Pitts [23]. The model of neuro nets caused the appearance of abstract automata theory, a theory which helps study the behavior and interaction of evolving systems independently of their structure. The Kleene-Glushkov algebra [13, 7]

is the main tool for the description of the behaviors of finite state systems. Automata theory originally concentrated on the study of analyses and synthesis problems, generalization of finite state automata and complexity. Interaction in explicit form appeared only in 70s as a general theory of interacting information processes. It includes the CCS (Calculus of Communicated Processes) [24, 25] and the π -calculus of R. Milner [26], CSP (Communicated Sequential Processes) of T. Hoare [10], ACP (Algebra of Communicated Processes) [3] and many other various branches of these basic theories. Now all these calculi and algebras are the basis for modern research in this area. Fairly complete survey of the classical process theory is presented in the Handbook of Process Algebras [4], published in 2001.

Insertion modeling is a trend that is developing over the last decade as an approach to a general theory of interaction of agents and environments in complex distributed multi-agent systems. The first works in this direction have been published in the middle of 90s [6, 15, 16]. In these studies, a model of interaction between agents and environments based on the notion of insertion function and the algebra of behaviors (similar to some kind of process algebra) has been proposed. The paradigm shift from computing to interaction was extensively discussed in computer science that time, and our work was in some sense a response to this trend. But the real roots of the insertion model should be sought even earlier, in a model of interacting of control and operational automata, proposed by V. Glushkov back in the 60s [8, 9] to describe the structure of computers. In the 70s the algebraic abstraction of this model were studied in the theory of discrete processors and provided a number of important results on the problem of equivalence of programs, their equivalent transformations and optimization. Macroconveyor models of parallel computing, which were investigated in 80s years [11], even more close to the model of interaction of agents and environments. In these models, the processes corresponding to the parallel processors can be considered as agents that interact in an environment of distributed data structures.

In recent years, insertion modeling has been applied to the development of systems for the verification of requirements and specifications of distributed interacting systems [2, 12, 19–21]. The system VRS, developed in order from Motorola, has been successfully applied to verify the requirements and specifications in the field of telecommunication systems, embedded systems, and real-time systems. A new insertion modeling system IMS [17], which is under development in the Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, is intended to extend the area of insertion modeling applications. We found many common features of the tools used in software development area based on formal methods and techniques used in biologically inspired cognitive architectures. This gives us hope to introduce some new ideas to the development of this subject domain.

This paper presents the main principals of insertion modeling and the conception of cognitive architecture based on insertion modeling. To understand the formal part of the paper reader must be familiar with the concepts of labeled

transition system, bisimilarity and basic notions of general process theory. The mathematical foundation of insertion modeling is presented in [18].

2 The Basic Principals

Insertion modeling deals with the construction of models and study the interaction of agents and environments in complex distributed multi-agent systems. Informally, the basic principles of the paradigm of insertion modeling can be formulated as follows.

1. The world is a hierarchy of environments and agents inserted into them.
2. Environments and agents are entities evolving in time.
3. Insertion of agent into environment changes the behavior of environment and produces new environment which is in general ready for the insertion of new agents.
4. Environments as agents can be inserted into higher level environment.
5. New agents can be inserted from external environment as well as from internal agents (environments).
6. Agents and environments can model another agents and environments on the different levels of abstraction.

All these principles can be formalized in terms of transition systems, behavior algebras, and insertion functions. This formalization can be used as high level abstractions of biological entities needed for computer modeling of human mind.

The first and the second principals are commonly used in information modelling of different kinds of systems, for example as in object oriented or agent programming. They are also resembling to M. Minsky's approach of the society of mind [27].

The third principal is clear intuitively, but has a special refinement in insertion modelling. We treat *agents* as transition systems with states considered up to bisimilarity (or up to behavior, which is the same). The *type of an agent* is the set of actions it can perform. The term *action* we use as a synonym of label for transitions, and it can denote signals or messages to send, events in which an agent can participate etc. This is the most general notion of agent which must be distinguished from more special notions of autonomous or intellectual agents in AI.

Transition system consists of states and transitions that connect states. Transitions are labeled by *actions* (signals, events, instructions, statements etc.). Transition systems are evolving in time changing their states, and actions are observable symbolic structures used for communication. We use the well-known notation $s \xrightarrow{a} s'$ to express the fact that transition system can evolve from the state s to s' performing action a . Usually transition systems are nondeterministic and there can be several transitions coming from the same state even labeled by the same action. If we abstract from the structure of states and concentrate only on (branching) sequences of observable actions we obtain the state equivalence

called *bisimilarity* (originated from [28] and [24], exact definition can be found in [18]). Bisimilar states generate the same behavior of transition systems.

Environment by definition is an agent that possesses the *insertion function*. Given the state of environment s and the state of agent u , insertion function computes the new state of environment which is denoted as $s[u]$. Note that we consider states up to bisimilarity and if we have some representation of behaviors, the behaviors of environment and agent can be used as states. The state $s[u]$ is a state of environment and we can use insertion function to insert a new agent v into environment $s[u] : (s[u])[v] = s[u, v]$. Repeating this construction we can obtain the state of environment $s[u_1, u_2, \dots]$ with several agents inserted into it. Insertion function can be considered as an operator over the states of environment, and if the states are identified with behaviors, then the insertion of a new agent changes the behavior of environment.

Environment is an agent with insertion function, so if we forget the insertion function, then environment can be inserted as agent into a higher level environment and we can obtain hierarchical structure like

$$s[s_1[u_{11}, u_{12}, \dots]_{E_1}, s_2[u_{21}, u_{22}, \dots]_{E_2}, \dots]_E$$

Here notation $s[u_1, u_2, \dots]_E$ explicitly shows the environment E to which the state s belongs (environment indexes can be omitted if they are known from the context). This refines the fourth principle.

Environment is an agent which can be inserted into external environment and having agents inserted into this environment. The evolution of agents can be defined by the rules for transitions. The rules $s[u] \xrightarrow{a} s[u, v]$ and $s[t[u, v]] \xrightarrow{a} s[t[u, v]]$ can be used for the illustration of the 5-th principal.

We consider the creating and manipulation of the models of external and internal environments as the main property of cognitive processes of intellectual agent. Formalization of this property in terms of insertion modeling supports the 6-th principal.

Cognitive architecture will be constructed as a multilevel insertion environment. Below we shall define the main kinds of blocks used for construction of cognitive architecture. They are *local description unites* and *insertion machines*.

3 Multilevel Environments

To represent behaviors of transition systems we use *behavior algebras* (a kind of process algebra). Behavior algebra is defined by the set of actions and the set of behaviors (processes). It has two operations and termination constants. Operations are prefixing $a.u$ (a - action, u - behavior) and nondeterministic choice $u + v$ (u and v - behaviors). Termination constants are successful termination Δ , deadlock 0 , and undefined behavior \perp . It has also approximation relation \sqsubseteq , which is a partial order with minimal element \perp , and is used for constructing a complete algebra with fixed point theorem. To define infinite behaviors we use equations in behavior algebra. These equations have the form of recursive definitions $u_i = F_i(u_1, u_2, \dots), i = 1, 2, \dots$ and define left hand side functions as

the components of a minimal fixed point. Left hand sides of these definitions can depend on parameters $u_i(x) = F_i(u, x)$ of different types. In complete behavior algebra each behavior has a representation (normal form)

$$u = \sum_{i \in I} a_i \cdot u_i + \varepsilon_i$$

which is defined uniquely (up to commutativity and associativity of nondeterministic choice), if all $a_i \cdot u_i$ are different (ε_u is a termination constant).

The *type of environment* is defined by two action sets: the set of environment actions and the set of agent actions. The last defines the type of agents which can be inserted into this environment: if the set of agent actions is included in the set of agent actions of environment then this agent can be inserted into this environment. This relation is called *compatibility* relation between agents and environments (agent is compatible with environment if it can be inserted into this environment). *Multilevel environment* is a family of environments with distinguished the most external environment. The compatibility relation on the set of environments defines a directed graph and we demand for multilevel environment that the outermost environment would be reachable from any environment of the family in this graph.

To define the insertion function for some environment it is sufficient to define transition relation for all states of environment including states with inserted agents. The common approach is to define behavior by means of rules. The following is an example of such rule:

$$\frac{s \xrightarrow{b} s', u \xrightarrow{a} u'}{s[u] \xrightarrow{c} s'[u']} P(a, b, c)$$

This rule can be interpreted as follows. Agent in the state u can make a transition $u \xrightarrow{a} u'$. Environment allows this transition if the predicate $P(a, b, c)$ is true. This rule defines behavior property of environment in some local neighborhood of the state $s[u]$. So such a rule belongs to the class of local description units discussed in the next section.

At a given moment of time an agent belongs (is inserted) to only one environment. But if the type of an agent is compatible with the type of another environment it can move to this environment. Such a movements can be described by the following types of rules:

$$\frac{u \xrightarrow{\text{moveup } E} u'}{E[F[u, v], w] \xrightarrow{\text{moveup}(F \rightarrow E)} E[F[v], u', w]} P_1(E, F, u, \text{moveup}(E))$$

moving from internal to external environment;

$$\frac{u \xrightarrow{\text{movedn } F} u'}{E[F[v], u, w] \xrightarrow{\text{movedn}(E \rightarrow F)} E[F[u', v], w]} P_2(E, F, u, \text{movedn}(F))$$

moving from external environment to internal one;

$$\frac{u \xrightarrow{\text{moveto } G} u'}{E[F[u, v], G[w]] \xrightarrow{\text{moveto}(F \rightarrow G)} E[F[v], G[u', w]]} P_3(E, F, u, \text{moveto}(F))$$

moving to another environment on the same level. In all cases permitting conditions must include the compatibility conditions for corresponding agents and environments. The rules above define the property of a system called mobility and underlies the calculus of mobile ambients of Luca Cardelli [5].

4 Local Description Units over Attribute Environments

A special type of environments is considered in cognitive architecture to have a sufficiently rich language for the description of environment states properties. These environments are called *attribute environments*. There are two kinds of attribute environments – *concrete* and *symbolic*.

The state of concrete attribute environment is the valuation of *attributes* - symbols that change their values while changing the state in time. Each attribute has type (numeric, symbolic, enumerated, agent and behavior types, functional types etc.). Some of functional and predicate symbols are interpreted symbols. Now logic formulas can be used for the description of properties of agent or environment states. We use the first order logic formulas as the basis that can be extended by fuzzy logic, temporal logic etc.

The general form of *local description unit* is the following:

$$\forall x(\alpha(x, r) \rightarrow \langle P(x, r) \rangle \beta(x, r)),$$

where x is a list of typed parameters, r is a list of attributes, $\alpha(x, r)$ and $\beta(x, r)$ are logic formulas, $\langle P(x, r) \rangle$ is a process - finite behavior of an environment. Local descriptions can be considered as formulas of dynamic logic, or Hoare triples, or productions - the most popular units of procedural memory in AI. In any case they describe local dynamic properties of environment behavior: for all possible values of parameters, if precondition is true then a process of a local description unit can start and after successful termination of this process a postcondition must be true.

The states of symbolic environment are formulas of basic logic language of environment. Such formulas are abstractions of classes of concrete states. Each symbolic state covers the set of concrete states and the traces generated by local description units cover the sets of concrete traces.

Local description units are the main units of knowledge representation in cognitive architecture. A set of local description units can be used for the definition of transitions of environment. In this case they can be considered as procedural knowledge units. Logic knowledge can be represented as environment with the states representing the current knowledge, and the local description units corresponding to the rules of inference in corresponding calculus. Local description units can be applied in forward and backward modes. Forward mode can be used for the generating of new knowledge, backward mode – for answering queries.

5 Insertion Machines

Another construction blocks for cognitive architecture are insertion machines intended for implementation of insertion environments. The input of insertion machine is the description of a multilevel environment (a model of an environment) and its initial state, an output depends on the goal that is put to machine.

Multilevel environments are represented in cognitive architecture by means of *environment descriptions* for different levels and a set of local description units for insertion functions. Environment description contains the signature of environment that includes types of attributes, types of inserted agents, and also the description of sets of environment and agent actions. Local description units used for the definition of insertion function are organized as a knowledge base with special data structures providing efficient access to the needed descriptions and history of their use.

To implement multilevel environment different kinds of insertion machines are used. But all of them have the general architecture represented on the Fig.1. Three main components of insertion machine are *model driver* (MD), *behavior*

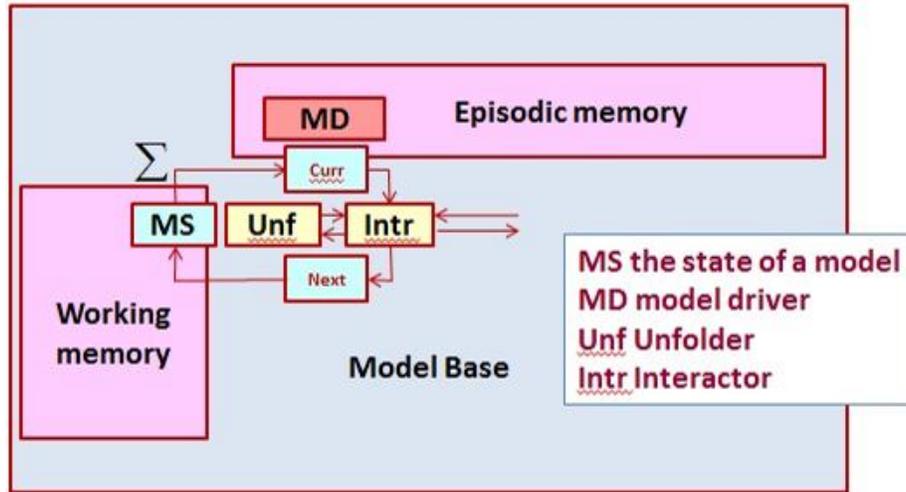


Fig. 1. Architecture of Insertion Machine

unfolder (Unf), and *interactor* (Intr). Model driver is a component which controls the machine traversal along the behavior tree of a model. The state of a model is represented as a text in the input language of insertion machine and is

considered as an algebraic expression. The input language includes the recursive definitions of agent behaviors, the notation for insertion function, and possibly some compositions for environment states. Before computing insertion function the state of a system must be represented in the form $s[u_1, u_2, \dots]$. This functionality is performed by agent behavior unfolders. To make the movement, the state of environment must be reduced to the normal form

$$\sum_{i \in I} a_i \cdot u_i + \varepsilon$$

where a_i are actions, u_i are environment states, ε is a termination constant. This functionality is performed by the module *environment interactor*. It computes the insertion function calling recursively if it is necessary the agent behavior unfolders.

Two kinds of insertion machines are distinguished: *real time* or *interactive* and *analytical* insertion machines. The first ones are functioning in the real or virtual environment, interacting with it in the real or virtual time. Analytical machines intended for model analysis, investigation of its properties, solving problems etc. The drivers for two kinds of machines correspondingly are also divided into interactive and analytical drivers. Interactive driver after normalizing the state of environment must select exactly one alternative and perform the action specified as a prefix of this alternative. Insertion machine with interactive driver operates as an agent inserted into external environment with insertion function defining the laws of functioning of this environment. External environment, for example, can change a behavior prefix of insertion machine according to their insertion function. Cognitive interactive driver has criteria of successful functioning in external environment, it accumulates the information about its past in episodic memory, develops the models of external environment, uses some learning algorithms to improve the strategy of selecting actions and increase the level of successful functioning. In addition it should have specialized tools for exchange the signals with external environment (for example, perception of visual or acoustical information, space movement etc.).

Analytical insertion machine as opposed to interactive one can consider different variants of making decisions about performed actions, returning to choice points (as in logic programming) and consider different paths in the behavior tree of a model. The model of a system can include the model of external environment of this system, and the driver performance depends on the goals of insertion machine. In the general case analytical machine solves the problems by search of states, having the corresponding properties (goal states) or states in which given safety properties are violated. The external environment for insertion machine can be represented by a user who interacts with insertion machine, sets problems, and controls the activity of insertion machine. Analytical machine enriched by logic and deductive tools are used for generating traces of symbolic models of systems. The state of symbolic model is represented by means of properties of the values of attributes rather than their concrete values.

Insertion machine with separated external environment interface can be implemented as a transition system with hidden structure that separates the kernel

environment state and the states of inserted agents. Such implementation can be more efficient and can be constructed using partial computations or other specialization and optimization programming tools.

6 Cognitive Architecture

Like well-known cognitive architectures such as Soar [14], ACT-R [1] or many other from the list of BICA society [29] insertion cognitive architecture ICAR is an environment for construction of cognitive agents. The main blocks of this architecture are local description units, agents, represented by their behaviors, and insertion machines. Building blocks are collected in memory units that have structures of knowledge bases or associative memories.

On the abstract level ICAR has the same architecture as cognitive agents that can be created in it. From this point of view it can be considered as an intellectual assistant for user who interacts with ICAR in the process of creating cognitive agents. The general architecture of cognitive agent of ICAR is represented on Fig.2.

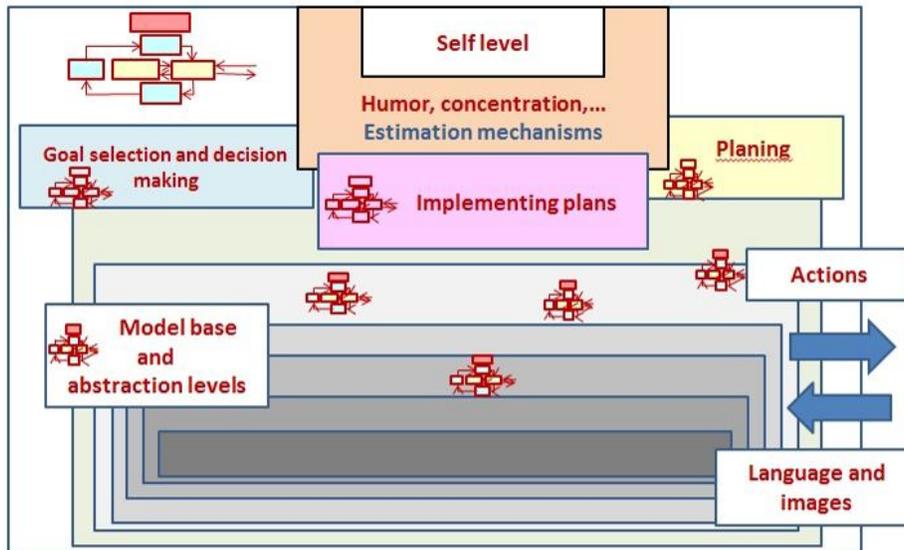


Fig. 2. Insertion cognitive architecture

In general cognitive agent is constructed as a real time multilevel insertion machine which realizes itself as a highest level internal environment. As an agent,

this machine is inserted in its external environment and has the means to interact with it. This external environment includes a user and objects of external (real or virtual) world to which agent has access.

One or several self-models can be inserted into the internal environment of cognitive agent to be used when interacting with external environment or making decisions and planning future activities. An agent has an estimation mechanism to evaluate the success of its behavior. This mechanism is realized in the form of a special agent that can observe the interaction of a system with external environment and make estimation according to some criteria. These criteria can be predefined initially and evolves in the future according to obtained experience. The main goal of a system is achieving maximum success repeated.

The self-models of cognitive agent are created and developed together with the models of external environment. If the external environment contains other agents, they can be modeled by internal environment which creates corresponding machines and interprets those machines using corresponding drivers, comparing the behaviors of models and external agents. All these models are evolving and developing in the process of accumulating the experience in interaction with the external world.

Some mechanisms that model emotional or psychological features (humor and concentration, pleasure and anger, etc.) can be implemented at higher levels of cognitive structure. The mechanisms of decision making, planning and executing plans are also at higher levels.

The main part of cognitive structure is the base of models describing the history of cognitive agent functioning at different levels of abstraction. The interface with external world provides language (symbolic) communication and image processing. All interaction histories are processed in the working memory of the self-level insertion machines and then transferred to the appropriate levels of a model base.

The model base is always active. The analytical insertion machines which control and manage the structure of model base are always busy with searching solution of problems and performing tasks with unsatisfactory answers, or creating new models. All this activity models subconscious levels of cognition and time-to-time interact with the higher levels of cognitive structure. Independent levels of cognitive structure are working in parallel.

The hierarchy of environments of cognitive agent in some sense are similar to six layers of neocortex. Moving from low levels to higher ones the levels of abstraction are increased and used more and more abstract symbolic models. How to create such models is a big challenge and we are working on it now.

Cognitive analytical insertion machines of ICAR are used by cognitive agents to learn their models and their interaction with external environment to solve problems better, accepts user helps as a teacher and teach user how to interact better with ICAR. General learning mechanisms are the parts of model drivers of different types.

7 Conclusions

The description of cognitive architecture in the last section is a very tentative reflection of our far goals. The nearer goals include the further development of our system of proving program correctness [22], communication in natural language, and living in virtual reality. As a zero approximation of ICAR the insertion modeling system [17] is used.

References

1. Anderson, J.R., Lebiere, C.: *The Atomic Components of Thought*. Mahwah: Lawrence Erlbaum Associates (1998)
2. Baranov, S., Jervis, C., Kotlyarov, V., Letichevsky, A. and Weigert, T.: *Leveraging UML to deliver correct telecom applications in UML for Real: Design of Embedded Real-Time Systems* by L.Lavagno, G. Martin, and B. Selic (editors), 323-342, Kluwer Academic Publishers (2003)
3. Bergstra, J. A. and Klop J. W.: Process algebra for synchronous communications. *Information and Control*, 60 (1/3), 109–137 (1984)
4. Bergstra, J. A., Ponce, A. and Smolka, S. A.(eds.): *Handbook of Process Algebra*. North-Holland (2001)
5. Cardelli, L. and Gordon, A. D.: Mobile ambients. In: *Foundations of Software Science and Computation Structures: First International Conference, FOSSACS '98*, Springer-Verlag (1998)
6. Gilbert, D. R. and Letichevsky, A. A.: A Universal Interpreter for Nondeterministic Concurrent Programming Languages. In: Gabbrielli, M. (Ed.) *Fifth Compu-log Network Area Meeting on Language Design and Semantic Analysis Methods*, September (1996)
7. Glushkov, V.M.: On an Algorithm of Abstract Automata Synthesis. *Ukrainian Mathematical Journal*, 12(2), 147–156 (1960).
8. Glushkov, V.M.: Automata Theory and Questions of Design Structure of Digital Machines. *Cybernetics* 1, 3–11 (1965)
9. Glushkov, V.M. and Letichevsky, A. A.: Theory of Algorithms and Discrete processors. In: Tou, J. T. (Ed.) *Advances in Information Systems Science*, vol. 1, Plenum Press, 1-58 (1969)
10. Hoare, C.A.R.: *Communicating Sequential Processes*. Prentice Hall (1985)
11. Kapitonova, J. and Letichevsky, A.: *Mathematical Theory of Computational Systems Design*. Moscow, Science (1988) (in Russian)
12. Kapitonova, J., Letichevsky, A., Volkov, V. and Weigert, T.: Validation of Embedded Systems. In: R. Zurawski (Ed.) *The Embedded Systems Handbook*, CRC Press, Miami (2005)
13. Kleene, S. C.: Representation of Events in Nerve Nets and Finite Automata. In: Shannon, C. E., McCarthy, J. (eds.) *Automata Studies*, Princeton University Press, pp. 3-42 (1956)
14. Laird, J. E., Newell, A., Rosenbloom, P. S.: SOAR: an Architecture for General Intelligence. *Artificial intelligence*, 33, 1–64 (1987)
15. Letichevsky, A. and Gilbert, D.: A general Theory of Action Languages. *Cybernetics and System Analyses*, 1 (1998)

16. Letichevsky, A. and Gilbert, D.: A Model for Interaction of Agents and Environments. In: Bert, D., Choppy, C. and Moses, P. (eds.) *Recent Trends in Algebraic Development Techniques*. LNCS, vol 1827, Springer Verlag (1999)
17. Letichevsky, A., Letychevskiy, O. and Peschanenko, V.: *Insertion Modeling System*. In: Proc. PSI 2011, LNCS, vol 7162, pp. 262–274, Springer Verlag (2011)
18. Letichevsky, A.: *Algebra of Behavior Transformations and its Applications*. In: Kudryavtsev, V. B. and Rosenberg, I.G. (eds.) *Structural Theory of Automata, Semigroups, and Universal Algebra*. NATO Science Series II. Mathematics, Physics and Chemistry, vol 207, pp. 241–272, Springer Verlag (2005)
19. Letichevsky, A., Kapitonova, J., Letichevsky, A. Jr., Volkov, V., Baranov, S., Kotlyarov, V. and Weigert, T.: *Basic Protocols, Message Sequence Charts, and the Verification of Requirements Specifications*. ISSRE 2004, WITUL (Workshop on Integrated reliability with Telecommunications and UML Languages) , Rennes, 4 November (2005)
20. Letichevsky, A., Kapitonova, J., Letichevsky, A. Jr., Volkov, V., Baranov, S., Kotlyarov, V. and Weigert, T.: *Basic Protocols, Message Sequence Charts, and the Verification of Requirements Specifications*. *Computer Networks*, 47, 662–675 (2005)
21. Letichevsky, A., Kapitonova, J., Letichevsky, A. Jr., Volkov, V., Baranov, S., Kotlyarov, V. and Weigert, T.: *System Specification with Basic Protocols*. *Cybernetics and System Analyses*, 4 (2005)
22. Letichevsky, A., Letichevsky, O., Morokhovets, M. and Peschanenko, V.: *System of Programs Proving*. In: Velichko, V., Volosin, A. and Markov, K. (eds.) *Problems of Computer Intellectualization*, Kyiv, V. M. Glushkov Institute of Cybernetics, pp.133–140 (2012)
23. McCulloch, W.S. and Pitts, W.: *A Logical Calculus of the Ideas Immanent in Nervous Activity*, *Bull. of Math Biophys.*, 5, 115–133 (1943)
24. Milner, R.: *A Calculus of Communicating Systems*, LNCS, vol 92, Springer Verlag (1980)
25. Milner, R.: *Communication and Concurrency*. Prentice Hall (1989)
26. Milner, R.: *The Polyadic π -calculus: a Tutorial*. Tech. Rep. ECS-LFCS-91-180, Laboratory for Foundations of Computer Science, Department of Computer Science, University of Edinburgh, UK (1991)
27. Minsky, M.: *The Society of Mind*. Touchstone Book (1988)
28. Park, D.: *Concurrency and Automata on Infinite Sequences*. LNCS, vol 104, Springer-Verlag (1981)
29. Samsonovich, A. V.: *Toward a Unified Catalog of Implemented Cognitive Architectures (Review)*. In: Samsonovich, A.V., Johansdottir, K.R., Chella, A. and Goertzel, B. (eds.) *Biologically Inspired Cognitive Architectures 2010: Proc. 1st Annual Meeting of BICA Society, Frontiers in Artificial Intelligence and Applications*, vol 221, pp. 195–244 (2010)

Part 1. Main ICTERI Conference

1.1 ICT Infrastructures, Integration and Interoperability

Wireframe Model for Simulating Quantum Information Processing Systems

Mizal Alobaidi¹, Andriy Batyiv², and Grygoriy Zholtkevych²

¹ Tikrit University,

Faculty of Computer Science and Mathematics, P.O. Box-42, Tikrit, Iraq

mizalobaidi@yahoo.com

² V.N. Karazin Kharkiv National University,

School of Mathematics and Mechanics, 4, Svobody Sqr., 61022, Kharkiv, Ukraine

{generatorglukoff,g.zholtkevych}@gmail.com

Abstract. This paper continues the papers series concerned with authors' research of quantum information processing systems based on the formal model of abstract quantum automata. The previous papers of the series were focused on mathematical modelling of quantum information processing systems. This paper describes the core components of information technology for studying the systems by computer simulation. A domain model and a behavioural model of the wireframe for simulating quantum information processing system are presented in the paper. The language "AQuanAut" for description abstract quantum automata is specified. The problems encountered during the realizing of the wireframe model are discussed.

Keywords. Simulation, simulation wireframe, domain model, behavioural model, quantum computation, finite-level quantum system, quantum information processing system, abstract quantum automaton

Key terms. MathematicalModel, ComputerSimulation, Specification-Process

1 Introduction

Necessity to increase processing power for computational devices, traffic capacity and security level for communication channels leads to new challenges in the fields of Information and Communication Technology (ICT). Nowadays quantum informatics is considered as an approach to meet the challenges. The research in the area of quantum informatics uses knowledge in the fields of mathematics, physics, and computer science. So, the corresponding research technique covers the wide variety of methods including both theoretical developments and experimental investigations. We should note that quantum experiments are quite difficult and expensive for accomplishing. Therefore, the problem to simulate such experiments arises naturally.

This paper describes an attempt to construct the wireframe model for simulating quantum information processing systems basing on the notion of an abstract quantum automaton [1, 2]. Taking in account that majority of specialists in computer science are not familiar with the notation and concepts of quantum informatics in detail authors have tried to provide maximal presentation completeness for the notation and basic concepts. As needed, one can use [5] for more deep acquaintance with problems and methods of quantum informatics.

2 Brief Survey of Physical Grounds and Mathematical Models

In the section the description of mathematical model for quantum information processing systems in the terms of abstract quantum automata and physical grounds for the model are given.

At the highest abstraction level an abstract quantum automaton can be considered as a two-component hybrid quantum-classical system $\mathfrak{A}(\mathcal{Q}, \mathcal{T})$. The quantum component \mathcal{Q} of the system functions as a memory and the classical component \mathcal{T} of the system implements an information process control. Interactions between the classical control component and the quantum memory sustain the integrity of the system.

Now let's suppose satisfiability a number of properties for systems \mathcal{Q} and \mathcal{T} .

Assumption 1 *The quantum memory \mathcal{Q} is an m -level quantum system.*

It means that an m -dimensional Hilbert space \mathcal{H}_m is associated with the quantum memory. This space is known as the state space. The memory is completely described by its pure state, which is a one-dimensional subspace of the state space. This subspace is uniquely represented by the ortho-projector $|\psi\rangle\langle\psi|$ on the unit vector $|\psi\rangle$ which generates the subspace.

In contrast to pure states mixed states are used to describe memory whose state is not completely known. Rather more detailed suppose we know that a memory is in one of a number of states $\{|\psi_k\rangle\langle\psi_k| : k = 1, \dots, s\}$ with respective probabilities $\{p_k : k = 1, \dots, s\}$. We shall call $\{p_k, |\psi_k\rangle\langle\psi_k| : k = 1, \dots, s\}$ an ensemble of pure states. The density operator for the system is defined by the equation $\rho = \sum_{k=1}^s p_k |\psi_k\rangle\langle\psi_k|$.

Mixed states are identified with density operators. The statement that pure states are described by one-dimensional ortho-projectors allows to consider pure states as indecomposable states.

Assumption 2 *The control system \mathcal{T} is a deterministic labelled transition system.*

It means that \mathcal{T} is a tuple $(C, n_*, T, A, \#, Trans, \text{dom}, \text{codom}, \lambda)$ where

- C is a finite set of computational nodes;
- n_* is some element of C , which is called the initial node;
- T is a finite set of terminal nodes such that $C \cap T = \emptyset$;

- A is a finite alphabet of possible outcomes;
- \sharp is some picked outcome in A ;
- $Trans$ is a finite set of transitions;
- $\text{dom} : Trans \rightarrow C$ maps each transition into the node, which is source of this transition;
- $\text{codom} : Trans \rightarrow N$ maps each transition into the node, which is sink of this transition, where $N = C \cup T$;
- λ is a map from $Trans$ onto A .

The following conditions should be satisfied for this tuple

- determinacy:** for any $\tau', \tau'' \in Trans$ equalities $\text{dom}(\tau') = \text{dom}(\tau'')$ and $\lambda(\tau') = \lambda(\tau'')$ imply $\tau' = \tau''$;
- default label:** for any $\tau \in Trans$ the equality $\lambda(\tau) = \sharp$ is equivalent to $\text{dom}^{-1}(\text{dom}(\tau)) = \{\tau\}$;
- reachability:** for any $n \in N$ there exists a finite sequence $\tau_1, \dots, \tau_k \in Trans$ such that $\text{dom}(\tau_1) = n_*$, $\text{codom}(\tau_k) = n$, and for all $j = 1, \dots, k-1$ the following equality $\text{codom}(\tau_j) = \text{dom}(\tau_{j+1})$ is true.

Assumption 3 *A snapshot of an abstract quantum automaton is completely described by the pair $|\psi\rangle, n$, where $|\psi\rangle \in \mathcal{H}_m$ is a unit vector represented the current memory state and $n \in N$ is some node of the control system.*

Assumption 4 *Each interaction between the memory and the control system is a pair consisting of a memory state transformation and a jump from one node to another: $|\psi\rangle, n \vdash |\psi'\rangle, n'$. This jump should be determined by a transition: if $|\psi\rangle, n \vdash |\psi'\rangle, n'$ then there exists the unique transition $\tau \in Trans$ realizing the jump, i.e. $\text{dom}(\tau) = n$ and $\text{codom}(\tau) = n'$. Such interactions are called quantum actions.*

Assump. 4 does not determine any algorithm for performing quantum actions. The next assumption describes explicitly such an algorithm for removing this defect. This description uses the notion of a generating isometric operator for quantum actions. The notion has introduced and studied in detail in [1].

Assumption 5 *The quantum action associated with a computational node n is described by its generating isometric operator $W_n : \mathcal{H}_m \otimes l^2(Out_n)$, where $Out_n = \lambda(\text{dom}^{-1}(n))$. This operator determines the interaction $|\psi\rangle, n \vdash |\psi'\rangle, n'$ by using the following procedure:*

1. *select randomly the label $a \in Out_n$ in accordance with the probability distribution*

$$\Pr(a | \psi) = \langle \psi | W_n^\dagger (\mathbf{1} \otimes |a\rangle\langle a|) W_n | \psi \rangle, \quad (1)$$

where W_n^\dagger is the adjoint operator to the operator W_n , $|a\rangle\langle a| = \delta(a, \cdot)$, and $\delta(\cdot, \cdot)$ is Kronecker delta;

2. *determine the transition $\tau \in Trans$ such that $\lambda(\tau) = a$ and $\text{dom}(\tau) = n$;*

3. compute the pair $|\psi'\rangle, n'$ by the following formulae

$$|\psi'\rangle = \frac{J(a)^\dagger W_n |\psi\rangle}{\sqrt{\Pr(a | \psi)}} \quad (2)$$

$$n' = \text{codom}(\tau) \quad (3)$$

where the isometric operator $J(a)$ acts from \mathcal{H}_m into $\mathcal{H}_m \otimes l^2(\Lambda)$ in compliance with the next formula $J(a)|\phi\rangle = |\phi\rangle \otimes |a\rangle$.

Thus, a trajectory of an abstract quantum automaton can be define as a sequence of interactions $|\psi_0\rangle, n_* \vdash |\psi_1\rangle, n_1 \vdash \dots \vdash |\psi_t\rangle, n_t$, where n_t is a terminal node. The corresponding sequence of labels $\lambda(\tau_1)\lambda(\tau_2)\dots\lambda(\tau_{t-1})$ such that $\text{dom}(\tau_1) = n_*$ and $\text{dom}(\tau_j) = n_j, \text{codom}(\tau_j) = n_{j+1}$, where $j = 1, \dots, t-1$, will be called an automaton trace. Stress that only an automaton trace is an observed part of the automaton trajectory.

3 Description of Simulation Wireframe Model

In the paper the Unified Modelling Language (UML) [6, 7] is used for specifying different details of quantum information processing systems. Object Constraint Language (OCL) expressions [3] are added to UML diagrams to describe a model precisely.

Describing the simulation wireframe model of abstract quantum automata we start with a specification of their composite structure (see Fig. 1). The component `memory` describes the quantum memory (the finite-level quantum system \mathcal{Q}) and the component `control` is a model for the classical control system \mathcal{T} .

The structural units of the component `control` describe three kinds of its constituents: nodes (N), transitions ($Trans$), and labels (Λ).

More ample description of an abstract quantum automaton structure is described by domain model (see Fig. 2). As it is shown in Fig. 2 the set of nodes is divided into two subsets of nodes: the subset of computational nodes and the subset of terminal nodes.

Computational nodes have two differences from the terminal nodes. Firstly, some quantum action is associated with each computational node in contrast to a terminal node. Secondly, a computational node has as minima one outgoing transition, whereas each terminal node has not any outgoing transition.

Properties of the control system for an abstract quantum automaton is set by Assump. 2. The static instance `default` of the class `Label` encapsulated into this class represents the specific label (so called the default label) for transitions that are determined uniquely by their sources and sinks. Existence of the default label is grounded by the "default label" condition. The determinacy condition of the control system can be described by the next constraint

```
context Transition inv: determinacy
  Transition::allInstances()->forAll(t1, t2: Transition |
    t1.dom = t2.dom and t1.tag = t2.tag implies
    t1.codom = t2.codom)
```

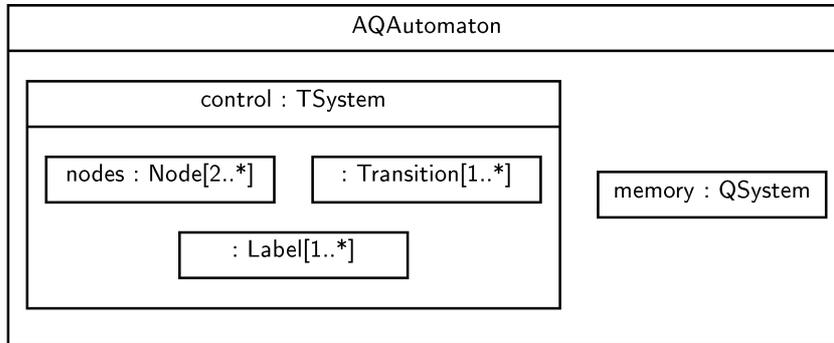


Fig. 1. Composite structure of an abstract quantum automaton

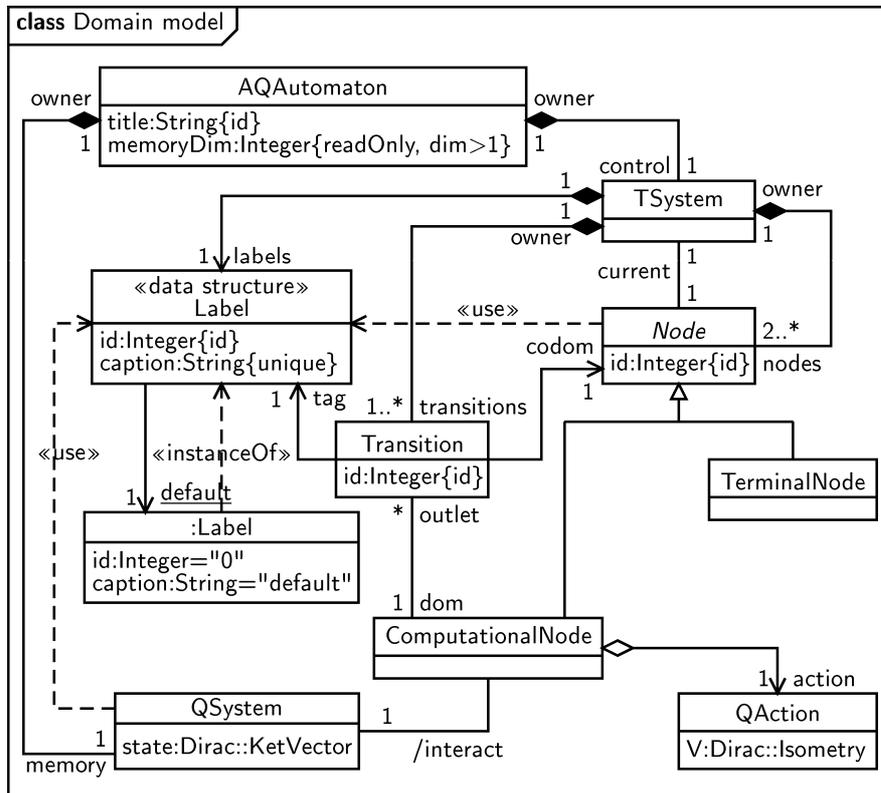


Fig. 2. Abstract quantum automaton domain model

Sustaining an interaction between the memory and the control system requires visibility of the component `memory` from the current computational node of the control system and vice versa. The derived association `interact` is intended to solve this task.

```
-- The definition of the association 'interact'
context ComputationalNode -- view from the control system
  def: interact: QSystem =
    self.current.owner.memory
context QSystem -- view from the memory
  def: interact: ComputationalNode =
    self.owner.control.current
```

The generalized model of an abstract quantum automaton behaviour is presented in Fig. 3.

This model specifies the behaviour of an abstract quantum automaton in compliance with [2, Def. 11]. The corresponding mathematical representation of all dynamical aspects for the interaction of automaton components are collected in Assump. 5.

The solution to use two concurrent threads for realizing the evolution of an automaton makes possibility to interrupt a simulation process. Necessity of the possibility has been established in the course of testing a trial implementation of the wireframe.

Initialization of an automaton is realized by the methods `TSystem::reset()` and `QSystem::set(state:Dirac::KetVector)`.

The method `QSystem::step(action:QAction)` implements a quantum character of the behaviour for the automaton. Detailed specification of the method is shown in Fig. 4. Each performance of this methods leads to changing the current computational node. Directly, changing of the current computational node is effected by the method `TSystem::setCurrent(outcome:Label)` specified by the next constraint

```
-- The rule for changing the current node
context TSystem::setCurrent(outcome:Label)
post: control =
  control@pre.transitions->any(tag = outcome).codom
```

This rule corresponds to Assump. 4 and formula (3) of Assump. 5.

The interaction shown in Fig. 4 implements one automaton jump (see Assump. 4).

Two methods of the class `QSystem` (`selectOutcomes(action:QAction)` and `getDistr(outs:Label[1..*],action:QAction)`) are used for building the probability distribution associated with the current memory state and the current action. Formula (1) of Assump. 5 is used to calculate this distribution.

Selection of one of the possible outcomes is effected by using the standard generator of random real numbers uniformly distributed in the segment $[0, 1]$.

Finally, formula (2) of Assump. 5 is applied to calculate new memory state under condition that outcome of the action is known.

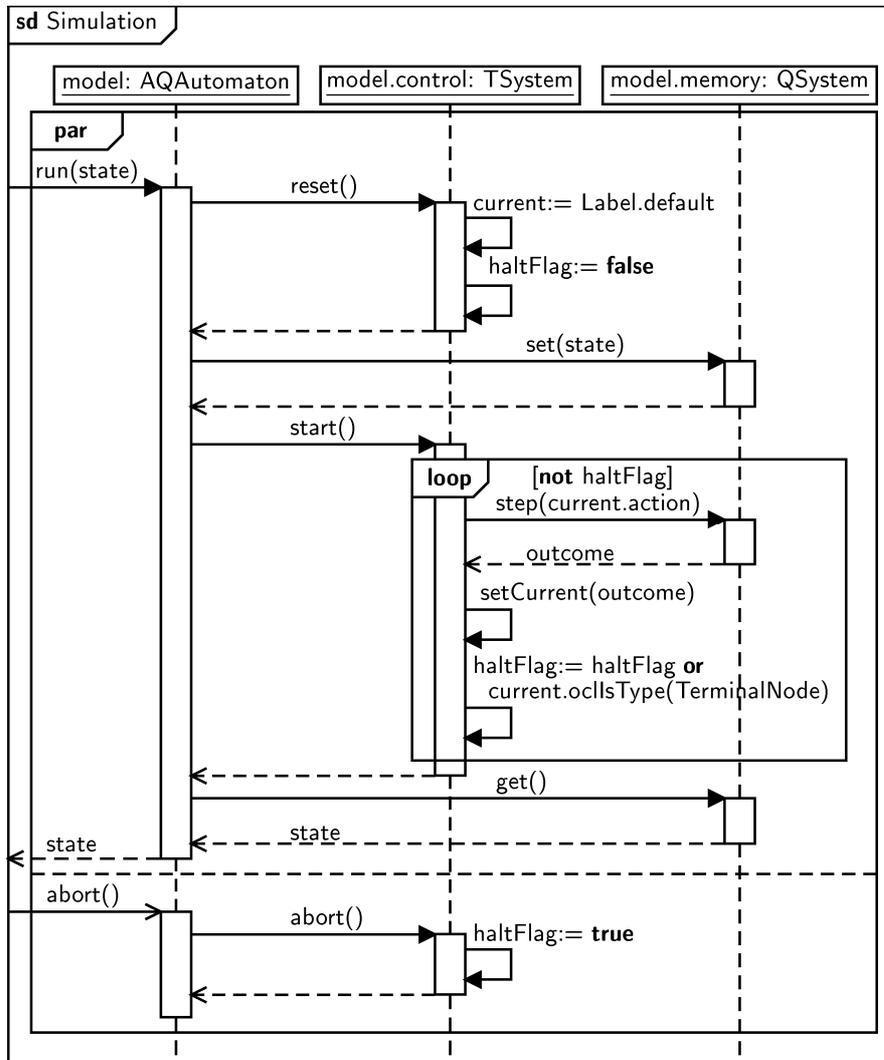


Fig. 3. Abstract quantum automaton behavioural model: interaction of components

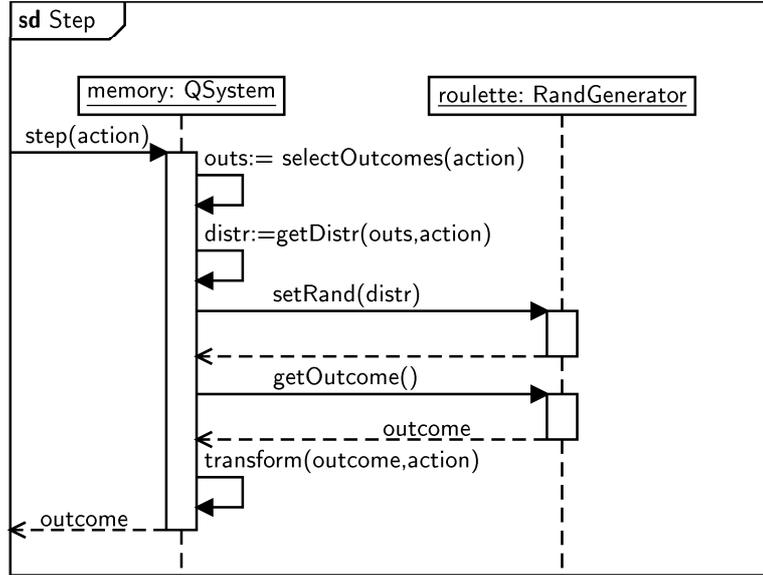


Fig. 4. Abstract quantum automaton behavioural model: step interaction

The label `outcome` returned by the method `step(QAction:action)` is used for changing the current node as it was discussed above.

4 Language "AQuanAut" for Specifying Abstract Quantum Automata

Language "AQuanAut" is a specification language for describing abstract quantum automata. Each AQuanAut-specification presents the description for the corresponding abstract quantum automaton. This description should provide sufficient data to build the programmed simulator for investigating the described automaton. For example, authors used these descriptions as input data for Java-application, which is a builder of abstract quantum automaton Java-simulators.

Syntax of language "AQuanAut" is below presented by use of Extended Backus - Naur Form [4].

The start symbol of "AQuanAut" grammar is denoted by `specification` in the paper. It is defined by the next rule, which fixes two-component structure of an abstract quantum automaton:

```

specification =
    'automaton', WS, identifier, EOL,
    'control:', EOL, control specification,
    'actions (memory levels number = ', levels number,
        '):', EOL, action specification,
    'end';
  
```

```
levels number = ? number of quantum memory levels ?;
```

The meta-identifier `identifier` denotes unique names of specifying objects and it is used for naming automata in the previous rule. It is defined by the following way:

```
letter = ? all upper-case and lower-case Latin letters ?;
digit = ? all decimal digits ?;
identifier = letter, {letter | digit};
```

The meta-identifiers `WS` and `EOL` are described two kinds of delimiters. They are determined by the following rules:

```
WS = ? all white space characters ?;
EOL = ? control sequence "new line" ?;
```

The next rules are used to define the meta-identifier `control specification`.

```
control specification =
    entry node specification,
    node specification, {node specification};
entry node specification = 'entry', WS, node specification;
```

The meta-identifier `node specification` is the main linguistic unit to specify the control system of an automaton. Each node is described by its identifier and by its outgoing transition list.

```
node specification =
    identifier, '(' , transition list, ')', EOL;
transition list =
    transition specification,
    {' , ' , WS, transition specification};
```

Each transition τ is described by its label $\lambda(\tau)$ and its sink node $\text{codom}(\tau)$.

```
transition specification = label, ':', WS, identifier;
```

Any identifier or the special symbol `"#"` can be used as a label. Note that the special symbol is used as the label `"default"` for nodes, which have only one outgoing transition.

```
label = '#' | identifier;
```

Now let's consider the last part of an automaton description, which is determined by the meta-identifier `action specification`.

Note that to determine an operator on a Hilbert space \mathcal{H}_m it is sufficient to specify its action on vectors from an ortho-normal basis, for example, $|0\rangle, \dots, |m-1\rangle$. So, we should specify vectors $|\Omega_0\rangle, \dots, |\Omega_{m-1}\rangle$ from $\mathcal{H}_m \otimes l^2(\Lambda)$. The number of memory levels m determined by the meta-identifier `levels number` (see the rule for definition the meta-identifier `specification` above).

Thus, the meta-identifier `action specification` is defined by the next way:

```

action specification =
    identifier, ':', action, EOL,
    {identifier, ':', action, EOL};

```

In this rule the meta-identifier `identifier` refers on a node identifier.

```

action =
    '[' , {index, ':', WS, 'S(', product-vector, ')', ', ', '},
    index, ':', WS, 'S(', product-vector, ')', ', '];
index = ? index of basis vector ?;

```

The meta-identifier `product-vector` is used for denoting the sum of elementary tensors, which corresponds to the image of the basis vector with index `index`.

```

product-vector =
    {index, ':', '|', function, '>', ', ', '},
    index, ':', '|', function, '>';
function = {label, ':', complex, ', ', '}, label, ':', complex;
complex = real | 'I', '*' , real | real '+', 'I', '*' , real;
real = ? real number ?;

```

Example 1. As example let's describe an automaton, which set a qubit in the state $|0\rangle$. Mathematical model for this automaton was described in [1, 2].

```

automaton QubitCleaner
control:
    entry measure(V0: exit, V1: flip)
    flip( #: exit)
actions (memory levels number = 2):
    measure: [0: S(0: |V0: 1>), 1: S(1: |V1: 1>)]
    flip: [0: S(1: |#: 1>), 1: S(0: |#: 1>)]
end

```

5 Trial Implementation of the Wireframe Model

Trial implementation of the model described above was performed by the authors and a group of Master students at the Department of Theoretical and Applied Computer Science at the V.N. Karazin Kharkiv National University.

As an implementation language was chosen language Java. This choice was due to the presence of a convenient free tool for rapidly develop compilers for Domain Specific Languages (ANTLR, see [8]) and a wide variety of free libraries for matrix calculations.

Table 1 shows libraries for matrix calculations, which were analysed in the process of working on the trial implementation.

Michael Thomas Flanagan Java Scientific Library was chosen for the trial implementation. This decision is grounded by the following constraints:

Table 1. Libraries for matrix calculations

Library Name	Library Location
COLT	http://acs.lbl.gov/software/colt/
Efficient Java Matrix Library (EJML)	http://code.google.com/efficient-java-matrix-library/
Java Matrix Library	http://jmatrices.sourceforge.net/index.html
Java Matrix Package (JAMA)	http://math.nist.gov/javanumerics/jama/
Michael Thomas Flanagan's Java Scientific Library	http://www.ee.ucl.ac.uk/~mflanaga/java/index.html
Universal Java Matrix Package	http://www.ujmp.org/

- the library should cover all matrix operations used under modelling abstract quantum automata;
- the library should provide interoperability with the library MPJ Express, which is an implementation of an MPI-like API used to write parallel Java applications for executing on a variety of parallel platforms ranging from multi-core processors to computing clusters/clouds [9].

The trial implementation of the wireframe model has revealed several problems, that need to be addressed:

1. processing time required to execute every step of quantum automaton has the exponential growth. Using parallel processing and grid computing can be considered as possible future solutions;
2. limited precision of computer processor may produce errors in the model of intermediate quantum memory states. Accumulation of these errors can be destroy correctness of physical postulates mapping. Application of symbolic computations can be considered as a possible future solution.

6 Conclusion

Summarising the above we can conclude:

- simulation wireframe model for studying quantum information processing systems is presented in the paper. This model is based on the notion of an abstract quantum automaton;
- the trial implementation of the wireframe model has performed. The realization detects a series of problems, which are described above;

- description language for quantum automaton "AQuanAut" has been developed.

Our nearest objective is re-engineering of the model to implement it basing on computing environment for high-performance computing and grid systems.

References

1. Alobaidi, M., Batyiv, A., Zholtkevych, G.: Abstract Quantum Automata as Formal Models of Quantum Information Processing Systems. In: V. Ermolayev et al.(eds.) ICT in Education, Research, and Industrial Applications. CCIS, vol. 347, pp. 19 – 38. Springer-Verlag, Berlin Heidelberg (2013)
2. Alobaidi, M., Batyiv, A., Zholtkevych, G.: Towards the Notion of an Abstract Quantum Automaton. arXiv:1204.3986v1 [cs.CC], <http://arxiv.org/abs/1204.3986>
3. Information technology – Object Management Group – Object Constraint Language (OCL). ISO/IEC 19507:2012(E)
4. Information technology – Syntactic metalanguage – Extended BNF. ISO/IEC 14977:1996(E)
5. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information, 10th Anniversary Edition. Cambridge University Press, Cambridge (2010)
6. OMG Unified Modeling LanguageTM (OMG UML), Infrastructure. OMG (2011), <http://www.omg.org/spec/UML/2.4.1/Infrastructure>
7. OMG Unified Modeling Language^{1M} (OMG UML), Superstructure. OMG (2011), <http://www.omg.org/spec/UML/2.4.1/Superstructure>
8. Parr, T.: The Definitive ANTLR Reference. Building Domain-Specific Languages. Pragmatic Bookshelf, Raleigh, NC Dallas, TX (2007)
9. Shafi, A., Carpenter, B., Baker, M.: Nested parallelism for multi-core HPC systems using Java. J. Par. Distr. Comp., vol. 69, 6, 532–545 (2009)

Modeling, Algorithms and Implementation of the Microcontroller Control System for the Ion Beam Forming Process for Nanostructures Etching

Aleksandr Ralo¹, Andrii Derevianko¹, Aleksandr Kropotov¹, Sergiy Styervoyedov¹
and Oleksiy Vozniy¹

¹ V.N. Karazin Kharkiv National University,

Svobody sq. 4, 61022, Kharkiv, Ukraine

ralo-a-n@mail.ru

Abstract. In this work a three-level control system for the vacuum-plasma system with the ion source based on high-frequency induction discharge architecture is described. The system is built on smart sensors and specially designed for operation in high EMI programmable logic controllers (PLCs) in the middle hierarchical level. The structure of PLC and the algorithms of the system are presented. Results of comparative simulation of classical management system and created one, as well as the results of the system applying to obtain beams of positive and negative ions and beams neutralized particles are reported.

Keywords. Programmable logic controller (PLC), control system, plasma technology, empirical model

Key terms. InformationTechnology, CommunicationTechnology, Software-System, Integration, Process

1 Introduction

Ion beams are an effective means of the surface treatment. Their application can be found in the industry of the integrated circuits manufacturing, as well as for research purposes as a powerful tool of micro-and nanoscale structures synthesis. However, during the materials treatment by ion beams due to effects of similar charges repulsion and the charge accumulation near the processed surface, defects are formed in the form of islet unetched films and vice versa – side etches. There are unwanted and unpredictable changes in the structure of the processed sample that can irreversibly change the characteristics of the ware [1]. There are several methods of excluding or compensating the charge accumulation. One of the most promising technique that can increase the rate of applicable products during plasma-beam processing is alternately etching by pulsed beams of ions of different signs and etching by high-energy beams

of neutral atoms, where the particle charge is neutralized by the ion beam away from the treated sample [2], [3], [4].

Short times and complex algorithms that should be taken into account during these experiments do not allow the operator to process without the use of modern computer automation systems that work using a process model. Therefore, the aim of this work was to create the intellectual management and control system, that will meet the requirements of operating conditions of the pulsed plasma-process plant for etching and micro- and nanostructures formation. An urgent task today is to build a system capable of independently analysis of historical data, and using them to provide the process control. Seeing the specifics of the pulsed plasma-ion process system must ensure the fastest possible response to the effects of interference that is impossible with the use of information-analytical systems (IAS) based on the classical scheme and used for the slower process.

2 The Use of Programmable Logic Controllers in Control Systems

Programmable logic controllers (PLCs) are widely used in such control systems as industrial automation and automation of scientific research. They provide significantly higher reliability than personal computers, and the same flexibility of working. Difficulties with changes in the operation make popular today microcontroller systems as not sufficient automation tool.

In order to improve the efficiency of the system it was proposed to build the hardware using three-tier architecture, as it shown in Fig. 1. The lower level consists of intelligent sensors and control elements, which control the PLC work.

PLCs have a much greater speed and are responsible for getting data from sensors, filtering, formation of data packets and for communicating with the control centre, created on the basis of a personal computer.

Algorithmic work of such system can be represented as follows:

1. Information about physical parameters, obtained by sensors, after conversion into digital codes enters with the corresponding interface to the programmable logic controller.
2. PLC forms an information packet, transmits the received information to the managing node (in our case, the system arbiter).
3. The central node addresses for the data needed to the historical data storage, analyzes them and, if necessary, generate management command. Information received from the PLC is also stored in the repository for further analysis.
4. This control command is received with the PLC, which transmits it to the appropriate control element.

This design exceeds the performance of classical one, because: first, damaged or incorrect data can be filtered by a PLC, and second, the PLC data is transferred using an optimum package format.

Modern PLC consists of three main parts:

- PLC processor module

- I/O modules
- Programming mechanism

Typically, these parts are combined with the use of crates on a physical level and a number of tiers on the logical one.

Despite the structural similarity of the PLC with the PC, the PLC must have some specific characteristics. For the experiments, and automation systems building using PLC, they should be able to work under different, sometimes very hard conditions, ensuring high availability.

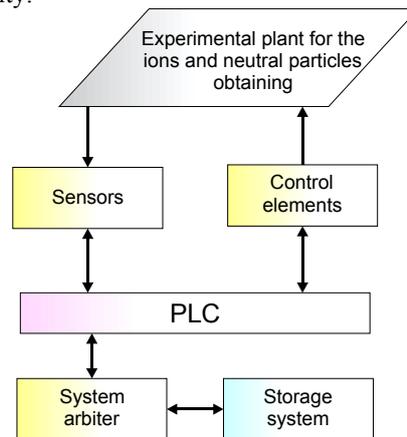


Fig. 1. The structure of the nodes communication using PLC

In the development of the PLC as a system bus selection was made from a list of the most popular, fully standardized well studied buses. The choice was made in favour of bus VMEbus [5].

By using this bus is possible to build scalable systems, where it can be increased both the number of modules that are responsible for input/output, and the number of processor modules, that makes it possible to distribute the control program in the case of its complexity.

Block diagram of the created PLC is shown in Fig. 2. As a PLC's control processor it has been selected 32-bit microcontroller AT91SAM7S from Atmel.

3 The Model Used

To provide a control on a given algorithm of complex technological processes, that includes the induction plasma source management, it is necessary to develop a model of the process. The best way to obtain it is to process the results of passive experiment that are stored in specially designed storage that provides quick access to historical data. After receiving the necessary amount of experimental data and the model creation, the control system can proceed in an active phase, i.e. manage the process.

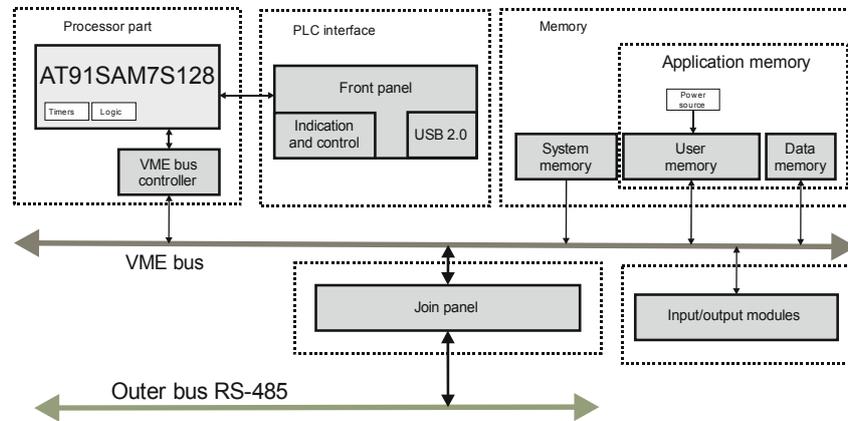


Fig. 2. PLC Block diagram

To construct the control system it was decided to use the model of the 5th class by Shantikumar [6]. In the proposed model, various simulation results for different initial conditions are used. In addition, on the basis of these results analytical model is generated. It is possible to identify the following properties of the model:

- Simulation model is used to determine the relationship between variable parameters and target factor
- An analytical model is a generalization of the results of various simulations; such systems are more accurate, than the solutions obtained only by using theoretical calculations, since they are based on multiple real data
- After receiving the analytical solution, which is not changing under the new simulations, this model can be used in the process control system to predict its results for the given parameters and, hence, it provides accurate information for quick decision-making [7]

The proposed class of models of simulation plays an important role in the investigation of the system behaviour. The results are used to derive analytical models to predict system performance. There may be a situation when the analytical model without simplifications can not be created, and the approximate model is not sufficiently accurate. Thus, models of this class are applicable in the cases:

- When the correlation between the target factor and the system parameters is not known that makes the analytical model is very difficult to develop, in this case, when analytical models are very expensive, unreliable or impractical, simulation can help in understanding the relationships between all factors that makes it possible to develop an analytical model
- In many practical problems the useful signal is often superimposed with lot of noise, which is almost impossible to take into account, simulation allows to investigate the behaviour of dynamic systems and to identify key parameters for evaluation, so that these parameters are then included in the analytical model

The process of a given class model constructing was described in [8]. As a result of simulation data processing system receives a functional relationship between the experiment factors and coefficients of these relationships. To find a list of dependencies

there has been used a list of 34 functions [9], that is sufficient to describe the most of physical processes.

4 Simulation of the Control System

During the system development phase to select the most efficient architecture of its work simulation was carried out. At the same time a number of software products used to data networks simulating was observed.

- OPNET and NetSim++ – graphical network modelling environment;
- SMURPH (University of Alberta) and Ptolemy (Berkeley) use an eclectic language for describing data lines;
- OMNet++ that uses its own language to describe the architecture that is then translated by the pre-processor to standard C++.

The choice was made in favour of the last instrument, as being open, easy embeddable in third-party software products and having multiple trusted code in its composition.

OMNet++ is not a programming language – it’s just a class library for the simulation. These classes are: modules, gateways, connections, settings, reports, histograms, assemblers, precision registers, etc.

First of all, it was composed a system model, built using “common bus” architecture with a central arbitration. The algorithm works as follows: arbiter sends a request to the sensors. The intensity of the queries depends on the particular sensor. The request is broadcast, thus it contains the sensor’s address. Sensor, recognizing the address, sends the data back. The size of datagram is also dependent on the sensor’s type.

The appearance of the simulation software modelling the system is shown Figure 3 (a).

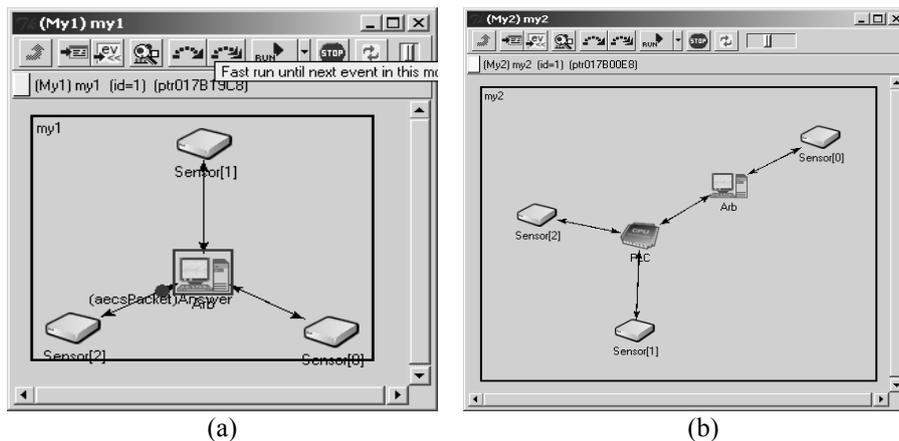


Fig. 3. OMNet++ work for simplified systems

To simulate the operation using the PLC part of the terminal devices was connected to an intermediate device, which operates as a PLC. The appearance of the resulting system in the program OMNet presented in Figure 3 (b).

Time and composition of each message sent and received is saved in a special vector file, that is then processed to obtain statistical information. As a result, basing on multiple system runs and changing the parameters such as: number of devices, data rate, processing time, etc. it was found that the use of PLC increases the productivity of an average of 1.5 times. Figure 4 shows the number of posts at different loads on the media for the cases of classical architecture and the use of the PLC. The fact that the use of OMNet++ allows programming in C++, made it possible to produce a bust of the parameters automatically.

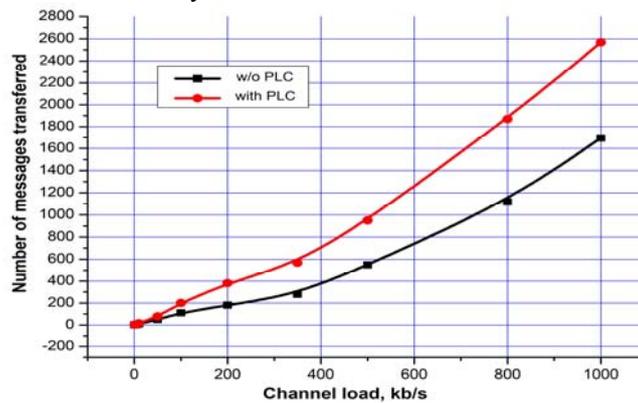


Fig. 4. Dependence of the sent messages number from the bus bandwidth

5 The Results of the Developed System Application

Concrete implementation of information management system was worked out for the experimental setup of vacuum-plasma nanostructures etching, schematically depicted in Figure 5 [10].

The vacuum chamber is a cylindrical volume of diameter 600 mm and length 800 mm. On the side flange of the chamber it was mounted an RF ion source based on the inductive discharge. With the ions extracted passage through the electrode system the beam of desired particle charge or neutral particles is formed. In front of the source a quadrupole mass spectrometer with an integrated energy analyzer with a resolution of 0.1 eV mounted. Plasma excitation is done with an inductor connected to the RF generator operating at a frequency of 13.56 MHz.

The management system's task was a search for optimal etching parameters and the stabilization of these parameters to ensure the repeatability of the experiment, as well as to change the programmable modes of control elements.

Modes of RF discharge source and power blocs forming voltages on the beam extraction system are controlled by PLC, that uses the internal bus commands to control elements, responsible for setting the pulse's amplitude, polarity, duration and duty

cycle. PLC via USB 2.0 was connected to the computer where the accumulation of experimental data and the search for functional relationships between them was done. The system supports the operator control mode and independent search of optimal parameters for this type of beams and maintains these parameters constant. The user interface and the corresponding signals taken from the outputs of power blocks are shown in Figure 6.

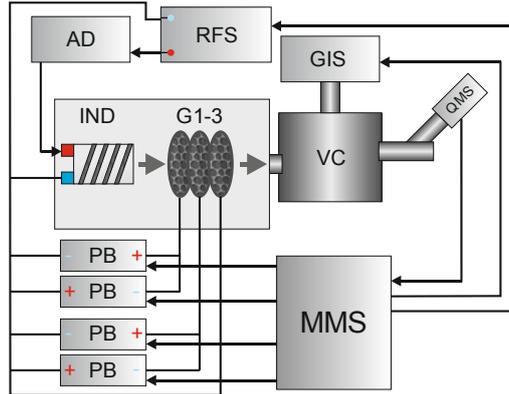


Fig. 5. Structural diagram of the setup for nanostructures etching. VC – vacuum chamber for ion-plasma treatment, QMS – quadrupole mass spectrometer with energy analyzer, RFS – RF source, AD – agreement device, PB – power blocks IND – inductor, G1-3 tree-grade electrode system, GIS – gas inlet system, MMS – microcontroller management system.

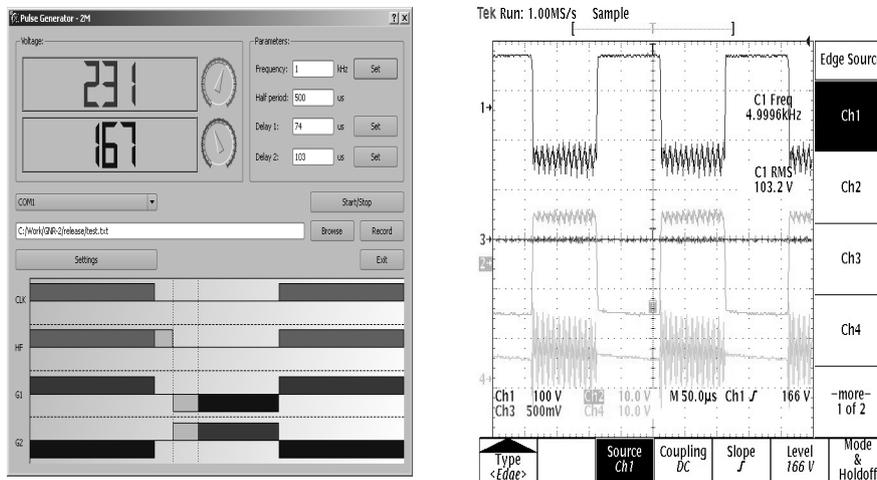


Fig. 6. The user interface and the corresponding control signals

6 Conclusion

Three-level information management system of vacuum-plasma system with the ion source based on high-frequency induction discharge is designed to produce beams of positive and negative ions and neutralized beam of particles used in the nanostructures etching. On the middle hierarchical level of the system there are specially designed programmable logic controllers based on a 32-bit microcontroller from Atmel AT91SAM7S, that have increased noise immunity characterized, advanced computing power and optimal capacity for a given task. PLCs make the primary information processing and greatly accelerate the process of information exchange, particularly at high pulse loads.

Despite the overall effectiveness of the method of constructing process models to manage them based on statistical data accumulated, in practice, large amounts of data to process and requirements for high-speed make this method not always usable. Therefore, the task of modelling and process control between the main control mechanism and the PLC allocation is important.

Testing of the system to manage the real process has shown its efficiency and allowed to obtain an adequate empirical model of the controlled ion source based on high-frequency discharge from tree-grades control.

References

1. Kinoshita, T., Hane, M., McVitte, J. P.: *Journal of Vacuum Science and Technology B* 14, 560 (1996)
2. Yunogami, T., Yokogawa, K., Mizutani, T.: Development of neutral-beam-assisted etcher. *Journal of Vacuum Science and Technology A* 13(3), 952 (1995)
3. Yokogawa, K., Yunogami, T., Mizutani, T.: Neutral- Beam-Assisted Etching System for Low-Damage SiO₂ Etching of 8-Inch Wafers. *Japan Journal of Application Physics* 35, 1901 (1996)
4. Vozniy, O. V., Yeom, G. Y.: High-energy negative ion beam obtained from pulsed inductively coupled plasma for charge-free etching process. *Appl. Phys. Lett.* 94, 231502 (2009)
5. VITA. Open standards, open markets, <http://www.vita.com> (2009)
6. Shanthikumar, J. G., Sargent, R. G.: Unifying view of hybrid simulation/analytic models and modeling. *Operations Research* 31 (6), 1030-1052 (1983)
7. Hsieh, T.: Hybrid analytic and simulation models for assembly line design and production planning. *Simulation Modeling Practice and Theory* 10, pp. 87-108 (2002)
8. Derevianko, A. V.: Constructing empirical models for the management of complex technological processes. *Bulletin of V.N. Karazin Kharkiv National University. No. 12: Mathematical modeling. Information technology. Automated control systems. № 863 - Kharkov: Publishing House of the KNU, P.101-110 (2009) (In Russian)*
9. Kuri-Morales, A., Rodriguez-Erazo, F.: A search space reduction methodology for data mining in large databases. *Engineering Application of Artificial Intelligence* 22, pp. 57-65 (2009)
10. Vozniy, O., Polozhiy, K., Yeom, G. Y.: *Journal of Application Physics* 102 083306 (2007)

Using Algebra-Algorithmic and Term Rewriting Tools for Developing Efficient Parallel Programs

Anatoliy Doroshenko¹, Kostiantyn Zhreb¹ and Olena Yatsenko¹

¹ Institute of Software Systems of National Academy of Sciences of Ukraine,
Glushkov prosp. 40, 03187 Kyiv, Ukraine
doroshenkoanatoliy2@gmail.com, zhreb@gmail.com, oayat@ukr.net

Abstract. An approach to program design and synthesis using algebra-algorithmic specifications and rewriting rules techniques is proposed. An algebra-algorithmic toolkit based on the approach allows building syntactically correct and easy-to-understand algorithm specifications. The term rewriting system supplements the algebra-algorithmic toolkit with facilities for transformation of the sequential and parallel algorithms, enabling their improvement.

Keywords. Algebra of algorithms, code generation, formalized design of programs, parallel computation, term rewriting

Key terms. FormalMethod, HighPerformanceComputing, ConcurrentComputation, Integration

1 Introduction

Nowadays uniprocessor systems are almost fully forced out by multiprocessor ones, as the latter allow getting the considerable increase of productivity of programs. Thus, the need of program parallelization arises [10]. There are libraries, such as pthreads, OpenMP, TBB and others [1], allowing developers to write parallel programs. Using these libraries a programmer manually divides code into independent sections, describes data exchange and synchronization between them. However, such method has substantial defects, in particular, related to committing of errors into program code and a time required for parallelization and debugging. Therefore, the parallelization process has to be automatized as much as possible, and in an ideal, should be carried out fully automatically, without participation of a programmer.

This paper continues our research on automation of process of designing and development of efficient parallel programs, started in [2], [9], [10], [11]. Our approach is based on usage of Integrated toolkit for Designing and Synthesis of programs (IDS) [2], [19]. The process of algorithm designing in IDS consists in the composition of reusable algorithmic components (language operations, basic operators and predicates), represented in Systems of Algorithmic Algebras (SAA) [2], [9], [19]. We used IDS for generation of sequential and parallel programs in Java and C++ on the basis

of high-level algorithm specifications (schemes). To automate the transformations of algorithms and programs we use term rewriting system Termware [8], [11]. The novelty of this paper is 1) adjusting IDS to generate parallel code in Cilk++ language, which is an extension to the C and C++ programming languages, designed for multi-threaded parallel computing [7] and 2) closer integration between IDS and Termware systems. The approach is illustrated on a recursive sorting algorithm (quick sort).

The problem of automated synthesis of program code from specifications has been studied extensively and many approaches have been proposed [13], [14]. Important aspects of program synthesis include 1) format of inputs (specifications), 2) methods for supporting concrete subject domains and 3) techniques for implementing transformation from specifications to output program code (these aspects roughly correspond to 3 dimensions of program synthesis discussed in [14]). For input specification, a popular option is using domain-specific languages (DSLs) [4], [17] that allow capturing requirements of subject domain. Other options include graphical modeling languages [5], [17], formal specification languages [16], ontologies [6] and algebraic specifications [3]. Using such formalisms enables analysis and verification of specifications and generated code. There are also approaches that provide specification not of program or algorithm, but of problem to be solved, in form of functional and non-functional constraints [18], examples of input/output pairs [15], or natural language descriptions [14].

Another crucial aspect of program synthesis is specialization for subject domain. Some approaches are restricted to a single domain, such as statistical data analysis [12] or mobile application development [17]; others provide facilities for changing domain-specific parts, by using ontological descriptions [6], grammars [16], or by providing generic framework that is complemented by domain-specific tools [18].

Finally, an important aspect is transformation from input specification into source code in a target language. A transformation algorithm can be hand-coded [12], but it reduces flexibility of system. Therefore, transformation is often described in a declarative form, such as rewriting rules [16], visualized graph transformations [17], code templates [6]. More complex approaches require searching the space of possible programs [18], possibly using genetic programming or machine learning approaches [14]. In [4], partial synthesis is proposed: generic parts of application are generated, and then completed with specific details manually.

In comparison, our approach uses algebraic specifications, based on Glushkov algebra of algorithms [2], but they can be represented in three equivalent forms: algebraic (formal language), natural-linguistic and graphical, therefore simplifying understanding of specifications and facilitating achievement of demanded program quality. Another advantage of IDS is a method of interactive design of syntactically correct algorithm specifications [2], [19], which eliminates syntax errors during construction of algorithm schemes. Specialization for subject domain is done by describing basic operators and predicates from this domain. Our approach uses code templates to specify implementations for operators and predicates; program transformations, such as from sequential to parallel algorithm, are implemented as rewriting rules. Such separation simplifies changing subject domain or transformations.

2 Formalized Design of Programs in IDS and Termware

The developed IDS toolkit is based on System of Algorithmic Algebras (SAA), which are used for formalized representation of algorithmic knowledge in a selected subject domain [2], [9], [19]. SAA is the two-based algebra $SAA = \langle \{U, B\}; \Omega \rangle$, where U is a set of logical conditions (predicates) and B is a set of operators, defined on an informational set; $\Omega = \Omega_1 \cup \Omega_2$ is the signature of operations consisting of the systems Ω_1 and Ω_2 of logical operations and operators respectively (these will be considered below). Operator representations of algorithms in SAA are called regular schemes. The algorithmic language SAA/1 [2] is based on mentioned algebra and is used to describe algorithms in a natural language form. The algorithms, represented in SAA/1, are called SAA schemes.

Operators and predicates can be basic or compound. The basic operator (predicate) is an operator (predicate), which is considered in SAA schemes as primary atomic abstraction. Compound operators are built from elementary ones by means of operations of sequential and parallel execution operators, branching and loops, and synchronizer `WAIT 'condition'` that delays the computation until the value of the `condition` is true (see also Table 1 in next section).

The advantage of using SAA schemes is the ability to describe algorithms in an easy-to-understand form facilitating achievement of demanded quality of programs. The IDS is intended for the interactive designing of schemes of algorithms in SAA and generating programs in target programming languages (Java, C++, Cilk++). In IDS algorithms are designed as syntactically correct programs ensuring the syntactical regularity of schemes. IDS integrates three forms of design-time representation of algorithms: regular schemes, SAA schemes (textual representation of SAA formulae) and flow graphs. For integration with Termware, in this paper IDS was also adjusted on generation of programs in Termware language.

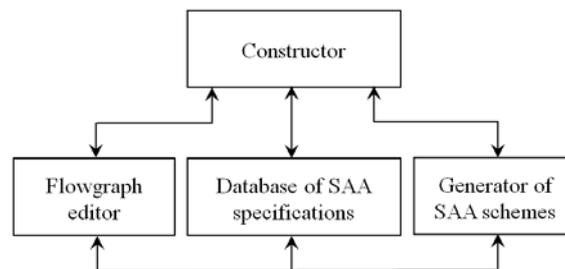


Fig. 1. Architecture of the IDS toolkit

The IDS toolkit consists of the following components (Fig. 1): constructor, intended for dialogue designing of syntactically correct sequential and concurrent algorithm schemes and generation of programs; flow graph editor; generator of SAA schemes on the basis of higher level schemes, called hyper-schemes [19]; and data-

base, containing the description of SAA operations, basic operators and predicates in three mentioned forms, and also their program implementations.

The constructor is intended to unfold designing of algorithm schemes by superposition of SAA language constructs, which a user chooses from a list of reusable components for construction of algorithms. The design process is represented by a tree of an algorithm [2], [19]. On each step of the design process the constructor allows the user to select only those operations, the insertion of which into the algorithm tree does not break the syntactical correctness of the scheme. The tree of algorithm constructing is then used for automatic generation of the text of SAA scheme, flow graph and the program code in a target programming language.

Example 1. We illustrate the use of SAA on Quicksort algorithm, which is given below in the form of SAA scheme. The identifiers of basic operators in the SAA scheme are written with double quotes and basic predicates are written with single quotes. Notice that identifiers can contain any text explaining the meaning of operator or predicate. It is not interpreted: it has to match exactly the specification in the database (however, since constructs are not entered manually, but selected from a list, the misspellings are prevented). The comments and implementations of compound operators and predicates in SAA schemes begin with a string of “=” characters.

```
SCHEME QUICKSORT_SEQUENTIAL =====
"main(n)"
===== Locals (
  "Declare an array (a) of type (int) and size (n)";
  "Declare a variable (i) of type (int)";
  "Declare a variable (end) of type (int)";
  "Fill the array (a) of size (n) with random
  values";
  "end := a + n";
  "qsort(a, end)";

"qsort(begin, end)"
===== IF NOT('begin = end')
  "Reduce (end) by (1)";
  "Reorder array (a) with range (begin) and (end)
  so that elements less than pivot (end) come
  before it and greater ones come after it; save
  pivot position to variable (middle)";
  "qsort(begin, middle)";
  "Increase (middle) by (1)";
  "Increase (end) by (1)";
  "qsort(middle, end)"
  END IF
END OF SCHEME QUICKSORT_SEQUENTIAL
```

To automate the transformation (e.g. parallelization) of programs we augment capabilities of IDS with rewriting rules technique [8], [11]. At the first step we construct high-level algebraic models of algorithms based on SAA in IDS (see also [2], [9], [19]). After high-level program model is created, we use parallelizing transformations to implement a parallel version of the program on a given platform (multicore in this paper). Transformations are represented as rewriting rules and therefore can be applied in automated manner. The declarative nature of rewriting technique simplifies adding new transformations. Also transformations are separated from language definitions (unlike approach used in [16]), therefore simplifying addition of new transformations or new languages.

We use the rewriting rules system Termware [8], [11]. Termware is used to describe transformations of *terms*, i.e. expressions in a form $f(t_1, \dots, t_n)$. Transformations are described as Termware *rules*, i.e. expressions of form `source [condition]-> destination [action]`. Here `source` is a source term (a pattern for match), `condition` is a condition of rule application, `destination` is a transformed term, `action` is additional action that is performed when rule fires. Each of 4 components can contain variables (denoted as `$var`), so that rules are more generally applicable. Components `condition` and `action` are optional. They can execute any procedural code, in particular use the additional data on the program.

3 Generation of Terms and Programs and Experimental Results

IDS system performs generation of programming code on the basis of an algorithm tree, received as a result of designing an algorithm in the IDS Constructor (see Section 2), and also code templates – implementations of basic operators and predicates in a target language (Java, C++, Cilk++), that are stored in IDS database. In the process of generation, IDS translates SAA operations into corresponding operators of programming language. Compound operators can be represented as subroutines (methods). IDS database contains various code patterns for generation of parallel programs, namely using WinAPI threads, Message Passing Interface (MPI), and Cilk++ operations [7]. For implementation of parallel version of our illustrative example (Quicksort algorithm), we used Cilk++ as it facilitates programming of recursive parallel programs [7]. Cilk++ is a general-purpose programming language, based on C/C++ and designed for multithreaded parallel computing.

Table 1 gives a list of main SAA operations and templates of their implementation in Termware and Cilk++, which are stored in the IDS database. The implementations contain placeholders like `^condition1^`, `^operator1^` etc., which are replaced with program code during the program generation.

For the purpose of transformation of some algorithm, IDS performs the generation of a corresponding term and developer specifies a set of rules for transformation. Then Termware carries out the actual transformation, the result of which can further be used for code generation in a programming language.

Table 1. The main SAA operations and templates of their implementation in Termware and Cilk++ languages

Text of SAA operation	Termware implementation	Cilk++ implementation
"operator1"; "operator2"	then (^operator1^, ^operator2^)	^operator1^; ^operator2^
IF 'condition' THEN "operator1" ELSE "operator2" END IF	IF (^condition1^, ^operator1^, ELSE (^operator2^))	if (^condition1^){ ^operator1^ } else {^operator2^}
FOR '(var) from (begin) to (end)' LOOP "operator1" END OF LOOP	FOR (%1, %2, %3, ^operator1^)	for (%1, %2, %3) { ^operator1^ }
("operator1" PARALLEL "opera- tor2")	Parallel(^operator1^, ^operator2^)	cilk_spawn ^operator1^; ^operator2^
WAIT 'condition'	WAIT (^condition1^)	cilk_sync;

Example 2. We will parallelize the sequential Quicksort algorithm (see Example 1), using IDS and Termware. For the parallelization, function `qsort` has to be transformed, so we generated the term for this function:

```
qsort(Params(begin, end),
      IF (NOT(Equal(begin, end)),
        then (Dec(end, 1),
              then (Partition(a, begin, end, end),
                    then (CALL(qsort(begin, middle)),
                          then (Inc(middle, 1),
                                then (Inc(end, 1),
                                      CALL (qsort(middle, end))))))))))
```

Then the operation of parallel execution of operations has to be added to this term. This is done by applying the following two Termware rules:

1. `then(CALL($x), then ($y, $z)) ->`
`Parallel (CALL($x), then($y, $z))`
2. `then($x1, Parallel($x2, $x3)) ->`
`then($x1, then(Parallel($x2, $x3),`
`WAIT(AllThreadsCompleted(n)))`

The first rule replaces the operation of sequential execution of operators with parallel execution. The second rule adds a synchronizer `WAIT(AllThreadsCompleted(n))`

pleted(n), which delays the computation until all threads complete their work. The result of the transformation is given below.

```

qsort(Params(begin, end),
IF(NOT(Equal(begin, end)),
  then (Dec(end, 1),
  then (Partition(a, begin, end, end),
  then (Parallel(
    CALL (qsort(begin, middle)),
    then (Inc(middle, 1),
    then (Inc(end, 1),
    CALL (qsort(middle, end))))),
    WAIT(AllThreadsCompleted(n))))))

```

Thus, as a result of parallelization, the first operator (thread) of `Parallel` operation executes the operator `qsort(begin, middle)`, and the second one calls two `Inc` operators and `qsort(middle, end)`. Operation `WAIT(AllThreadsCompleted(n))` performs the synchronization of threads. The threads are created recursively; their quantity is specified as an input parameter of function `main`. Notice that these transformations are only valid if two `qsort` calls are independent. The system doesn't check this property: it has to be asserted by a developer.

The resulting parallel algorithm scheme Quicksort was used for generation of code in Cilk++ using IDS system. The parallel program was executed on Intel Core 2 Quad CPU, 2.51 GHz, Windows XP machine. Fig. 2 shows the program execution time in seconds. The speedup at execution of program with usage of 2, 3 and 4 processors was 2; 2.9 and 3.8 accordingly, which shows that the program has a good degree of parallelism and is scalable.

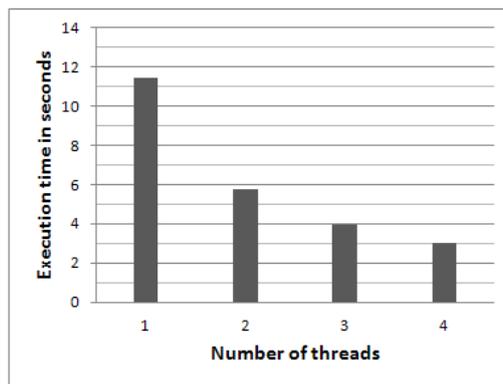


Fig. 2. The execution time of parallel Quicksort program on a quad-core processor; the size of input array is $5 \cdot 10^7$ elements

4 Conclusion

We have described our approach of constructing efficient parallel programs using high-level algebra-algorithmic specifications and rewriting rules technique. Algebra-algorithmic toolkit IDS and rewriting rules engine Termware are combined to enable formal, yet easy-to-understand algorithm specifications and automate program synthesis and parallelization process. The combined development toolkit can be retargeted to various subject domains and implementation languages, as exemplified by Cilk++. The developed system could be further extended with automated code analysis facilities based on rewriting technique.

References

1. Akhter, S., Roberts, J.: Multi-Core Programming. Intel Press, Hillsboro (2006)
2. Andon, F. I., Doroshenko, A. Y., Tseytlin, G. O., Yatsenko, O. A.: Algebra-Algorithmic Models and Methods of Parallel Programming. Akademperiodika, Kyiv (2007) (in Russian)
3. Apel, S. et al.: An Algebraic Foundation for Automatic Feature-Based Program Synthesis. *Science of Computer Programming*. 75(11), 1022–1047 (2010)
4. Bagheri, H., Sullivan, K.: Pol: Specification-Driven Synthesis of Architectural Code Frameworks for Platform-Based Applications. In: Proc. 11th Int. Conf on Generative Programming and Component Engineering, pp. 93–102, ACM, New York (2012)
5. Batory, D.: Program Refactoring, Program Synthesis, and Model-Driven Development. In: Proc. 16th Int. Conf. on Compiler Construction. LNCS 4420, pp. 156–171 Springer-Verlag, Berlin Heidelberg (2007)
6. Bures, T. et al.: The Role of Ontologies in Schema-Based Program Synthesis. In: Proc. Workshop on Ontologies as Software Engineering Artifacts, Vancouver (2004)
7. Cilk Home Page, <http://cilkplus.org/>
8. Doroshenko A., Shevchenko R.: A Rewriting Framework for Rule-Based Programming Dynamic Applications, *Fundamenta Informaticae*, 72(1–3), 95–108 (2006)
9. Doroshenko, A., Tseytlin, G., Yatsenko, O., Zachariya, L.: A Theory of Clones and Formalized Design of Programs. In: Proc. Int. Workshop on Concurrency, Specification and Programming (CS&P'2006), pp. 328–339, Wandlitz, Germany (2006)
10. Doroshenko, A. Y., Zhreb, K. A., Yatsenko, Ye. A.: On Complexity and Coordination of Computation in Multithreaded Programs. *Problems in Programming*, 2, 41–55 (2007) (in Russian)
11. Doroshenko, A., Zhreb, K.: Parallelizing Legacy Fortran Programs Using Rewriting Rules Technique and Algebraic Program Models. In: Ermolayev, V. et al. (eds.) *ICT in Education, Research, and Industrial Applications*. CCIS 347, pp. 39–59. Springer Verlag, Berlin Heidelberg (2013)
12. Fischer, B., Schumann, J.: AutoBayes: a System for Generating Data Analysis Programs from Statistical Models. *J. Funct. Program.* 13(3), 483–508 (2003)
13. Flener, P.: Achievements and Prospects of Program Synthesis. In: Kakas, A. C., Sadri, F. (eds.) *Computational Logic: Logic Programming and Beyond*. LNCS 2407, pp. 310–346, Springer Verlag, London (2002)

14. Gulwani, S.: Dimensions in Program Synthesis. In: 12th Int. ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming, pp. 13–24. ACM, New York (2010)
15. Kitzelmann, E.: Inductive Programming: a Survey of Program Synthesis Techniques. Approaches and Applications of Inductive Programming, LNCS 5812, pp. 50–73. Springer Verlag, Berlin Heidelberg (2010)
16. Leonard, E. I., Heitmeyer, C. L.: Automatic Program Generation from Formal Specifications using APTS. In: Automatic Program Development. A Tribute to Robert Paige, pp. 93–113. Springer Science, Dordrecht (2008)
17. Mannadiar, R., Vangheluwe, H.: Modular Synthesis of Mobile Device Applications from Domain-Specific Models. In: Proc. 7th Int. Workshop on Model-Based Methodologies for Pervasive and Embedded Software, pp. 21–28. ACM, New York (2010)
18. Srivastava, S., Gulwani, S., Foster, J. S.: From Program Verification to Program Synthesis. In: Proc. 37th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, pp. 313–326. ACM, New York (2010)
19. Yatsenko, O.: On Parameter-Driven Generation of Algorithm Schemes. In: Proc. Int. Workshop on Concurrency: Specification and Programming (CS&P'2012), pp. 428–438, Berlin, Germany (2012)

1.2 Machine Intelligence, Knowledge

Engineering and Management for ICT

An Intelligent Approach to Increase Efficiency of IT-Service Management Systems: University Case-Study

Nikolay Tkachuk¹, Vladyslav Sokol¹ and Kateryna Glukhovtsova¹

¹National Technical University “Kharkov Polytechnic Institute”
Frunze str., 21, Kharkov, Ukraine

tka@kpi.kharkov.ua, vladislav.sokol@gmail.com, kat_1109@mail.ru

Abstract. A comprehensive framework to increase efficiency of IT-services management systems (ITSMS) is proposed, which resolves 3 interconnected tasks in a target organization: 1) providing an effective configuration of ITSM-modules according to their specific features and needs; 2) integration a given ITSMS with existing enterprise architecture; 3) advanced incidents management in ITSMS. The applicability of this approach was tested successfully on the case-study at the National Technical University “Kharkov Polytechnic Institute” (www.kpi.kharkov.ua).

Keywords. IT-service management, effectiveness, multi-criteria ranking, data integration, adaptive ontology, case-based reasoning, e-learning

Key terms. Academia, ICTInfrastructure, KnowledgeManagementProcess, Model, SoftwareEngineeringProcess

1 Introduction: Problem Actuality and Research Objectives

Nowadays the concept of ITIL (IT Infrastructure Library) [1] and the new kind of computerized management systems, namely: IT Service Management Systems (ITSMS) became a growing and perspective approach to solve very important and complex technical problem and, at the same time, business-focused one: how to organize a well-structured and controllable IT-environment at an appropriate organization?

According to ISO/IEC 20000 [2] an IT Service Management System (ITSMS) provides “...a framework to enable the effective management and implementation of all IT-services”. Due to high complex and multi-dimensional nature of IT-services in large modern business organizations, which ITSMS are dealing with, recent publications in these domain present some sophisticated approaches to design and to use these facilities. One such important topic in ITIL-ITSM domain is the integration of ITSMS functionality into enterprise architecture (see, e.g. in [3,4]). Another recent trend in ITSMS-development is the usage of ontologies and model-driven architecture

(MDA) [5, 6] for knowledge handling and re-using. Their authors emphasize the actual need to elaborate and to apply several knowledge-oriented approaches to requirements analysis within ITSMS-development, and to quantitative quality assessment of appropriate project solutions.

Taking into account some ITSMS-issues mentioned above, the main objective of the research presented in this paper is to propose the first vision for intelligent complex approach to increase efficiency of typical ITSMS, with a proof of concept basing on the ITSMS university case-study. The rest of this paper is organized in the following way: Section 2 analyses some existing ITSMS, introduces our vision about their typical functionality, and shows the list of prioritized tasks to be resolved to increase an efficiency of an ITSMS. In Section 3 we present the method elaborated for effective ITSM-modules configuring in a target business organization, and Section 4 reports the first version of ITSMS - ontologies to integrate the selected modules into enterprise architecture (EA). In Section 5 the designing perspective for the combination case-based reasoning (CBR) with ontology-based approach to advanced incident management in ITSMS is briefly outlined. In Section 6 we present the university case-study for our method to estimate an effectiveness of different ITSMS configurations and discuss the results achieved. In Section 7 the paper concludes with a short summary and with an outlook on the next steps to be done in the proposed development framework.

2 Typical Functionality of ITSMS and the Complex of Intelligent Tasks to Increase its Efficiency

In order to elaborate a way how to provide a complex approach to increase an efficiency of ITSM-system operating, it is necessary to understand its typical functionality and to analyze its specific features.

2.1 Overview of existing ITSMS

We have analyzed some already existing ITSMS [7-10], and the results of this study is presented in the Table 1. Basically, all such systems can be divided into 3 groups, namely: (a) advanced business ITSM-products; (b) open source ITSM-solutions; (c) bespoke ITSM-systems.

To the group (a) belong such systems as, e.g., HP OpenView Service Desk [7] and BMC Remedy [8]. The first software product is the absolutely leader in this market segment, because the most part of organizations which prefer ITSM-business solutions from the group (a), are using exactly HP-platform. The number of its running installations is essentially less than HP, at least because of more expensive costs of Remedy ITSM Suite.

Table 1. Results of comparison for some ITSMS

<i>Criteria / Systems</i>	<i>BMC Remedy ITSM Suite 7.5</i>	<i>Axios Assyst 7.5</i>	<i>HP Service Manager 7.10</i>	<i>OMNINET OmniTracker ITSM Center 2.0</i>
Basic functionality	5	5	5	4
Maintainability	5	4	5	4
Report generation	4	5	5	4
Scaleability	4	2	3	5
Web-interface	5	5	5	5

ITSMS-solutions from the group (b) also are used in practice, but they definitely have limited functionality and provide less level of IT-services management. The typical open source ITSMS are, for instance, GLPI [8], OTRS [9], and some others, which are listed at the Web-resource SourceForge [10].

And, objectively, the business organizations, which are not ready to buy advanced software products from group (a), and which are not satisfied with functionality provided by ITSMS-systems from group (b), because they have some specific IT-needs and challenges, exactly these companies try to develop their own ITSMS-solutions to be considered as members of the group (c). The more detailed comprehensive study of some existing ITSMS-systems is presented in [11].

2.2 Typical ITSMS-functionality

Based on the given analysis of the real ITSMS (see above), we have elaborated the following vision for their typical functionality (see Fig. 1).

There are 5 main subsystems (or packages) of system functions, namely:

1. *IT Business Alignment*: this subsystem is supposed to implement a IT-strategy in given business organization with respect to its main goals and needs, and to provide a base for costs assessment to whole IT-infrastructure;
2. *Service Operations*: this facility is responsible for customer's requests management (regarding to a current incident and to a related problem), and for providing of ITSMS-support functions;
3. *Service Delivery Assurance*: this functional package implements a configuration and change management of all ITSMS-software tools thus is extremely important for a stable IT-environment;
4. *Service Design and Management*: this ITSMS-functionality provides detailed information about new perspective IT-services to be designed with respect to their availability and quality for IT-customers;
5. *Service Development and Deployment*: this subsystem allows to create and to test new ITSMS-services and appropriate IT-infrastructure solutions, including installa-

tion of new hard-ware components, development of additional software applications, and training programs for ITSM-staff and for end-users as well.

As we can see on the structure presented in Fig.1, each of these 5 subsystems is built from several functional modules (they are depicted as UML-classes). The most important of them are the following ones:

1. *Module M1 = "Incident management"*: it includes organizational procedures and appropriate tools to resolve current incidents, which IT-service users are facing with (hard-and software errors, network connection problems, request for consultations, etc.);

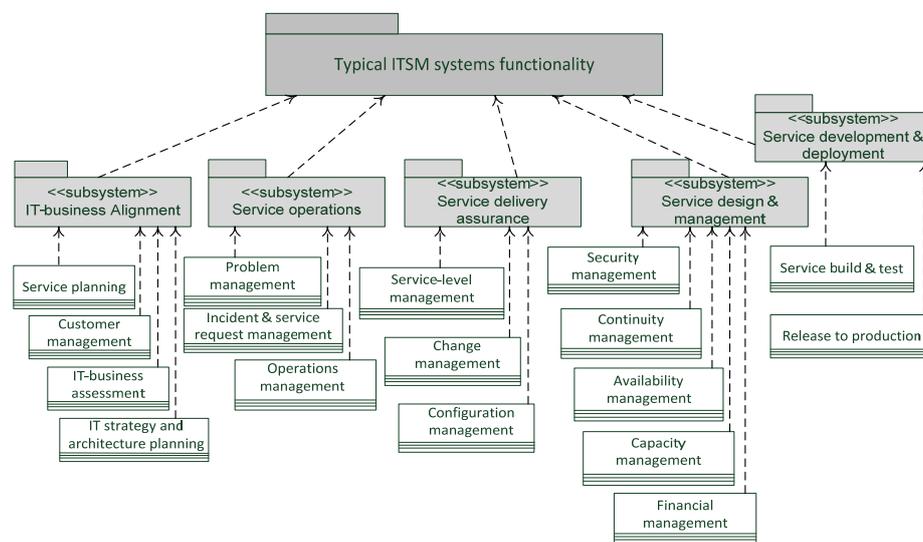


Fig. 1. Typical functionality of a ITSMS

- *Module M2 = "Problem management"*: this facility provides tools to detect and to eliminate any problem situation which is a reason for different incidents;
- *Module M3 = "Configuration management"*: this module supports all operating sub-schemes in the IT-infrastructure of given business organization;
- *Module M4 = "Change management"*: it supervises and coordinates all changes which arise in IT-infrastructure; .
- *Module M5 = "Service level management"*: this unit is responsible for definition and implementation of an appropriate level of IT-services to be provided for customers.

In ITIL-best practice manuals (see e.g. in [12]) the following 3 main schemes are considered to introduce these modules into IT-infrastructure of a target organization: a classic scheme (S1); a contract scheme (S2); an infrastructure-centered scheme (S3).

A *classic scheme* S1 is the most applied solution in the ITSM-domain, and it supposes the following sequence of modules M1-M5:

$$S1 = (M1, M3, M4, M2, M5) \quad (1)$$

This approach quickly allows to resolve the most actual communication problems between IT-service department and customers basing on incident management (module M1), and it provides some tools for all IT-services support (the modules M3 and M4), and after that a platform for future IT-infrastructure development is introduced (modules M2 and M5 respectively). But in this case it has to be taken into account this scheme is a most expensive way for a given business organization, and it requires a lot of resources exactly at an initial phase of whole ITSM-configuring framework.

A *contract scheme* S2 actually aims to formalize a communication process between IT-service department and customers, and it has the following modules-workflow:

$$S2 = (M5, M3, M1, M4, M2) \quad (2)$$

In this case all customer requirements to IT-services have to be collected and specified (in module M5), and appropriate IT-infrastructure sub-schemes can be built (using module M3), in order to define prospective IT-strategy in the target organization, next an operative ITSM-functionality is provided, including incident management (in module M1), change management (in module M4), and problem management (in module M2). Obviously, this scheme definitely has some risk factors regarding its efficiency, if the initial IT-service specifications were done not correctly (in module M5).

And, finally, an *infrastructure-centered scheme* S3 proposes the modules sequence indicated as following:

$$S3 = (M3, M4, M2, M1, M5) \quad (3)$$

that is, firstly, to provide tools for all IT-services support (modules M3 and M4 respectively). Secondly, this approach allows to manage all typical problem situations (in module M2), and already based on this one to detect and to resolve corresponded incidents by IT-service customers (in module M1). Thirdly, it creates an opportunity to define in computer-aided way the necessary composition and the IT-service level management (in module M5).

It is necessary to note that besides some empirical recommendations concerning the possible ITSM-modules configurations defined as (1)-(3), in the appropriate technical documentation there are no more or less proved suggestions about possible quantitative estimations for effectiveness of these alternative approaches.

2.3 The complex of intelligent tasks to increase of ITSM-system efficiency

Taking into account the results of performed analysis (see above), and based on some modern trends in the domain of ITSMS-development (see Section 1), the following list of prioritized tasks can be composed in order to increase ITSMS-efficiency, namely

1. to provide *an effective configuring* of ITSM-modules for a target organization, taking into account its specific features and needs;
2. to elaborate *an integration framework* for a given ITSM-system's configuration and for an existing enterprise architecture (EA);
3. to support *an advanced incidents management* in the already installed ITSM-system.

In our opinion, the task (I) can be resolved basing on some expert methods for multi-criteria ranking, with respect to specific IT-infrastructure's features and customer needs in a concerned business organization [13,14]. The task (II) belongs to already well-known integration issues in distributed heterogeneous information systems, and e.g. an ontology-based approach can be used for this purpose (e.g. in [3,6,15]). And, finally, to solve the task (III) an additional decision-making functionality for typical ITSM-services (see Fig.1) has to be elaborated, e.g. basing on the combination of case-based reasoning (CBR) approach with ontologies [16,17]. Below these tasks and their possible solutions are presented and discussed in more detail.

3 The Method for Effectiveness Estimation of Alternative ITSM-Module Configurations

To formalize the task (I) from their list considered in the Section 2.3, namely: to *provide an effective configuring* of ITSM-modules for a target business organization, the following factors have to taken into account: such a problem has a high complexity grade and it is semi-formalized; estimation criteria for it are of different nature and they are multi-valued; an information base to solve this task mainly can be collected basing on expert data only; available expert data could be quantitative and qualitative values both.

To solve this task we have chosen one of the multi-criteria ranking methods, which is presented in [14]. Accordingly to this approach the following steps have to be performed:

Step 1. A set of possible alternatives,

$$X = \{x_1, x_2, \dots, x_n\} = \{x_i, i = \overline{1, n}\} \quad (4)$$

and a set of global importance criteria to characterize these alternatives

$$K = \{K_1, K_2, \dots, K_m\} = \{K_j, j = \overline{1, m}\} \quad (5)$$

have to be defined.

Step 2. Each global criteria K_j is characterized by a subset of appropriate local criteria

$$K_j = \{k_{j1}, k_{j2}, \dots, k_{jq}\} = \{k_{jq}, q = \overline{1, Q}\} \quad (6)$$

further, a set of membership functions according to all local criteria alternatives

$$\{\varphi_{k_{j1}}(x_i), \varphi_{k_{j2}}(x_i), \dots, \varphi_{k_{jq}}(x_i)\} = \{\varphi_{k_{jq}}(x_i), q = \overline{1, Q}, j = \overline{1, m}\} \tag{7}$$

and the weight coefficients of their relative importance for these local criteria

$$\{w_{j1}, w_{j2}, \dots, w_{jq}\} = \{w_{jq}, q = \overline{1, Q}\} \tag{8}$$

have to be determined, where the following condition has to be fulfilled

$$\sum_{q=1}^Q w_{jq} = 1 \tag{9}$$

Step 3. To determine membership functions of alternatives $\{x_i, i = \overline{1, n}\}$ to criteria $K_j, \{j = \overline{1, m}\}$ based on an additive convolution of their local criteria

$$\varphi_{k_j}(x_i) = \sum_{q=1}^Q w_{jq} \varphi_{k_{jq}}(x_i) \tag{10}$$

Table 2. Definition of membership functions for criteria to alternatives (fragment)

Alternatives		Criteria K						
		K_1			...	K_M		
		k_{11}	...	k_{1Q}	...	k_{M1}	...	k_{Mm}
X	x_1	$\varphi_{k_{11}}(x_1)$...	$\varphi_{k_{1Q}}(x_1)$...	$\varphi_{k_{M1}}(x_1)$...	$\varphi_{Mm}(x_1)$

	x_n	$\varphi_{k_{11}}(x_n)$...	$\varphi_{k_{1Q}}(x_n)$...	$\varphi_{k_{M1}}(x_n)$...	$\varphi_{Mm}(x_n)$
	

Step 4. Taking into account the membership functions obtained $\{\varphi_{K_j}(x_i), j = \overline{1, m}\}$ for all alternatives $x_i, \{i = \overline{1, n}\}$ it is possible to determine a joined membership function for a generalized criterion K :

$$\varphi_K(x_i) = \sum_{j=1}^m w_j \varphi_{K_j}(x_i) \tag{11}$$

where $w_j, j = \overline{1, m}$ are coefficients of their relative importance $K_j, j = \overline{1, m}$.

Step 5. Finally, an alternative with a maximum value of membership function for generalized criterion K can be chosen as a target solution:

$$\varphi(x^*) = \max \{\varphi_K(x_i), i = \overline{1, n}\} \tag{12}$$

Below in Section 6 we present the case-study, which was performed to prove this method, and we discuss the results achieved.

4 Ontological Specifications for ITSMS-EA Integration Framework

As already mentioned above (see Section 2), any ITSMS has to be integrated into an existing EA of a target organization. In our approach this task (II) has to be resolved for an ITSMS-configuration defined with the method presented in Section 3.

This issue is already discussed intensively in a lot of publications, and their authors consider both its conceptual and technological aspects. E.g., an ITSMS-EA integration based on well-known SOA – framework is presented in [3], and as the important conceptual input for this issue the appropriate meta-model (actually, some kind of a domain ontology) for IT services is designed. In [5] an approach to integration of ITSM-services and business processes in given organization is elaborated, using ontological specifications to formalize the good practice guidance for ITSM. An ontology-based framework to integration of software development and ITSMS-functioning is proposed in [15], thus resulting in enhanced semantic-aware support tools for both processes. Even this brief overview allows us to conclude that exactly an ontology-based approach is a most effective way to solve this problem. That is why, in our opinion, to provide ITSM-EA integration effectively, it is necessary to combine the following information resources (IR), namely: a) IR related to ITSMS – functionality, b) IR concerned EA-domain, c) IR characterized a target organization (TO), which is facing an ITSMS-EA integration problem with. Let's define these IR (a)-(c) as: *Onto-ITSMS*, *Onto-EA*, and *Onto-TO* respectively. Thus, the IR needed to provide an ITSMS-EA integration should be specified using an appropriate joined ontology, designated as *Onto_ITSMS-EA*.

$$Onto_ITSMS - EA = \langle Onto - ITSMS, Onto - EA, Onto - TO \rangle \quad (13)$$

Obviously, some already existing ITIL / ITSM ontological specifications can be used for this purpose, e.g.: *Onto-ITIL* ontology elaborated in [5] basing on *OpenCyc* ontology (www.opencyc.org), *Onto-SPEM* (Software Process Engineering Meta-model) ontology [18], and *Onto-WF* (WorkFlow) ontology [19]. Taking these resources into account, we can represent the ontological specification for *Onto_ITSMS* in the following way

$$Onto - ITSMS = \langle Onto - ITIL, Onto - SPEM, Onto - WF \rangle \quad (14)$$

There are also several ontologies developed to specify EA, and according to one of recent and comprehensive researches in this domain presented in [20], we accept the following 3-level definition for EA-ontology

$$Onto - EA = \langle Onto - BT, Onto - AC, Onto - RS \rangle \quad (15)$$

where: *Onto_BT* is a sub-ontology of *Business Terms (BT)*, *Onto-SC* is a sub-ontology of *Architecture Components (AC)*, and *Onto-RS* as a sub-ontology of *RelationShips (RS)* among items of AC.

And finally, to define an *Onto-TO* ontology for target organization given in expression (13), its specific features and needs related to ITSMS-usage within

existing EA have to be taken into account. As a small excerpt of such domain-specific *Onto-TO*, which is elaborated in our University-ITSMS case-study (see Section 6), the following UML-class diagram in Fig. 2 is shown.

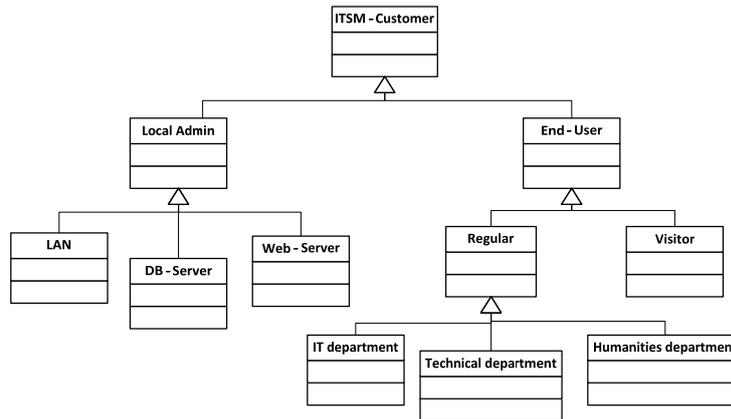


Fig. 2. Taxonomy of customers in a University-ITSMS as a part of a *Onto-TO* ontology

The proposed ontology-based approach for ITSMS-EA integration can also be used to elaborate the solution for the task (III) from their list completed in Section 2.3.

5 Adaptive Onto-CBR Approach to Advanced Incidents Management in ITSM

In order to solve the task (III), namely: to provide *an advanced incidents management* in ITSMS, accordingly to our inter-disciplinary vision about the ITSMS-development in general, we propose to amalgamate the following design-principles (i)-(iv) listed below

- (i) an *incident management* as a weak-formalized and complex task within the ITSMS-support for its customers can effective be resolved using one of the intelligent decision-support methods, e.g., using *CBR-method*;
- (ii) to enhance a CBR-functionality, especially with respect to specific needs in a target organization, an appropriate *domain-ontology* should be elaborated and used combining with CBR;
- (iii) because of the permanent changes in an IT-infrastructure of a given organization, and of the changes arising in its environment as well, such a domain-ontology has to be constructed as an *adaptive ontology*;
- (iv) to provide a possibility for knowledge gathering and their reusing in ITSMS, some *e-Learning models and technologies* can be applied.

There are already the approaches elaborated to combine a CBR-method with ontologies [16, 17], which allow to provide more efficiently a case-representation, to enhance case-similarity assessment, and to perform case-adaptation process for a new solution. From the other hand, an ontology-centered design for ITSM-services, and

especially, for *Incident Management (IM)*, is also discussed in some recent publications in this domain. In particular, the proposed in [21] *Onto-IM* ontology is built according to ISO/IEC20000 for ITIL/ITSM [2], it includes such concepts as *Incident Management*, *Incident Record*, *Incident Entity*, etc. specified using OWL (Ontology Web Language), and the small example of its notation is shown in Fig.3.

```
(contains some CreateIncidentRecord)
and (contains some
CreateIncidentRecordCapability)
and (contains some IncidentRecord)
and (contains some
IncidentRecordStructurePolicy)
and (contains some RecordIncident)
and (contains some ServiceDeskEmployee)
```

Fig. 3. The excerpt of Onto-IM ontology elaborated in [21]

These results provide a solution for the tasks (i)-(ii), but in our opinion to cover the task (iii) in efficient way, with respect to permanent changes in IT-infrastructure of a target organization, an appropriate ontology has to be constructed as an *adaptive* facility [22]. In this way the *Onto-TO* ontology given in Section 4 should be given as the following tuple

$$Onto - TO^{(adapt)} = \langle C, R, P, W^{(C)}, W^{(R)} \rangle \quad (16)$$

where, additionally to the basic components of any ontology, namely: C – set of concepts, R – a set of relationships among these concepts, and P – a set of axioms (semantic rules), the following ones have to be defined: $W^{(C)}$ is a set of weight coefficients for concepts of C , and a $W^{(R)}$ is a set of weight coefficients for relationships of R respectively. Usage of these weight coefficients allows us, e.g., to take into account an appropriate importance grade in several types of ITSMS-customers (see Fig. 2) to provide IM - services for them.

In order to get all information resources needed for a completed solution of tasks (i)-(iv), we propose to apply some *e-Learning* models and technologies within an ITSMS, especially, for skills training and experience gathering by ITSMS-staff, designated in the *Onto-IM* ontology as *Incident Manager*, *ServiceDeskEmployee*, *Specialist* [21]. For this purpose an e-Learning ontology (*Onto-EL*) can be used, e.g., in [23] the *Onto-EL* is elaborated to build for learners their personal paths in e-learning environment, according to the selected *curriculum* (*Incident Management* in terms of *Onto-IM*), *syllabus* (*Incident Record*) and *subject* (*Incident Entity*).

Summarizing aforementioned issues concerning the tasks listed in (i)-(iv), the conceptual mechanism to provide an *advanced incidents management* (AIM) in ITSMS can be represented at the high-architectural level as the UML-package diagram shown in Fig.4.

Below the approach to resolve the task (1) from their list given in Section 2 is illustrated using the real case-study within our research and practice activities to apply ITSMS to manage IT-infrastructure of National Technical University “Kharkov Polytechnic Institute” (www.kharkov.ua) referred in following as NTU “KhPI”.

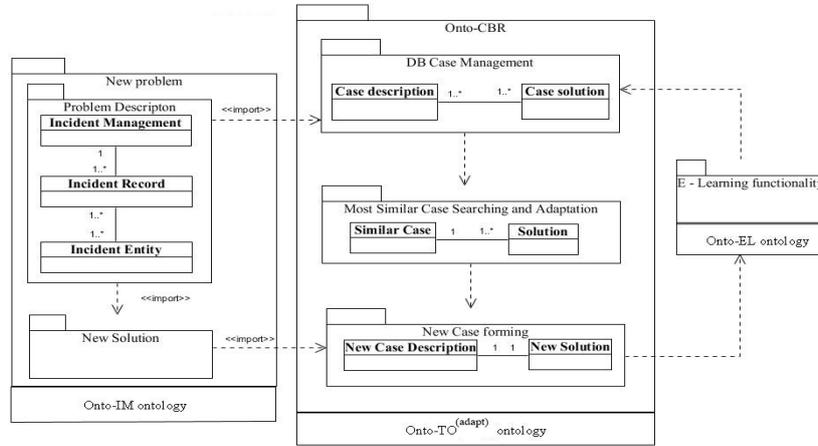


Fig. 4. The AIM – architectural framework (to compare with the scheme given in [24])

6 University Case-Study: Effective ITSM-Modules Configuring

It is to mention that exactly university- and /or a campus-domains are considered from many authors as a suitable example of ITSMS-usage (see, e.g., in [25-26]), because intensive research- and educational activities obviously require a modern and well-organized IT-environment. That is why we also proved our approach to effectiveness estimation of alternative configurations of ITSM-modules using the test-case data collected at the NTU “KhPI”.

6.1 Application domain description: IT-infrastructure of NTU “KhPI”

The NTU “KhPI” is one of the largest technical universities of Ukraine located in the city of Kharkiv, which is the important industrial and cultural center at the East of the country. The university has about 22000 students, ca. 3500 of faculty members, and accordingly there is the advanced IT-infrastructure to support all educational and research tasks. The main characteristics of IT-operating at the NTU “KhPI” are summarized in Table 3.

Table 3. Some technical data about IT-infrastructure NTU “KhPI”

Parameters	Values
PCs in the network configuration	1525
User’s accounts	2700
Buildings	23
Servers	60
Routers	80
Peripheral units	6000
IT-specialists in the central office	11
Incidences per day (registered)	5-7

In cooperation with the IT-staff at the University IT control office we have analyzed retrospective data about some typical problem situations occurred, and about the corresponded incidents, which daily have been resolved within the direct communication with IT-service customers. In this way the main types of ITSM-incidents and their initial problem situations were identified, and they are described in Table 4.

Table 4. Main types of ITSM-incidents and their related problem situations

№	Incident type	Cause (problem situation)
1	No Internet-connection at Dept or on local PC	- router was turned off; - network cable broken or failure on router hardware; - incorrect network setup; - problems with software on local PC
№	Incident type	Cause (problem situation)
2	High-loading of PC processor with a small number of active user's programs	- computer viruses - high degree of PC hard driver's de-fragmentation.
3	Installing problems for new software	- computer viruses - absence of additional (middleware) software needed for installation.
4	Failure to send email	-incorrect setup of local network server (proxy) -problems with central e-mail server.
5	Troubles in the use of third-party software	-lack of specific configuration, - improper use of system services.

Basing on the analysis results obtained, we can apply the elaborated method to estimate alternative ITSMS-module configurations (see Section 3).

6.2 Customizing of the elaborated estimation method: alternative configurations and criteria definition

According to the *Step 1* of the method presented in Section 2.2, the list of alternative ITSM-module configurations have to be defined, and in our case they are presented:

- X_1 = Service Desk subsystem (SDS) and Incident Management Module
- X_2 = SDS, Incident Management Module and Configuration Management Module
- X_3 = SDS, Incident Management Module and Change Management module
- X_4 = SDS, Incident Management Module and Problem Management Module

On the next *Step 2*, according to the formulas (4) - (10), we determine the criteria for the quantitative evaluation of the proposed alternatives and their performance indicators, which are shown in Table.5. These criteria and their indicators (metrics) are taken from [35], and they are recommended to evaluate effectiveness of IT-infrastructure in any business organization.

Table 5. List of values for global and local criteria (fragment)

Global and local criteria	Semantics performance measurement criteria and target values	Insecure value	Effective value	Scope of values
K_1	Effectiveness of incident management			
k_{11}	Average time incident resolution →min	>30 min.	15 min.	9999min.
k_{12}	Percentage of incidents resolved proactively →max	0%	15%	0-100%
Global and local criteria	Semantics performance measurement criteria and target values	Insecure value	Effective value	Scope of values
k_{13}	Percentage of incidents resolved at the first level of support →max	<65%	85%	0-100%
k_{14}	Percentage of incidents that have been resolver from the first time →max	<75%	90%	0-100%
K_2	Effectiveness of problem management			
k_{21}	The ratio of the number of solved problems to total problems (%)→max	<10%	35%	0-100%
.....			

For example, a value of 10 for an alternative X_3 to criteria k_{14} (see Table 5) means, that the implementation of *Service Desk* and *Incident Management Module* will help to increase the ratio of incidents, which are resolved successfully, to its effective value of 90%, etc. The obtained in this way results are given in Table 6.

Table 6. Estimated values for the alternatives with respect to the defined criteria (fragment)

	K_1 : Effective incident management → opt			
	k_{11} (opt =20M)	k_{12} (15%)	k_{13} (85%)	k_{14} (90%)
X_1	5	5	5	6
X_2	6	7	6	6
X_3	5	5	5	6
X_4	7	6	8	7
...			
...	

In order to implement the elaborated method with customized data introduced above, the special software tool was developed.

6.3 Results of estimation and their analysis

To continue the usage of our method presented in Section 2.2 (*Step 3* and *Step 4* respectively) using the pair-wise comparison the weight coefficients of relative impor-

tance (WCRI designated as $w(k_{i,j})$) for the local criteria regarding their global ones were determined:

- The WCRI values of the local criteria for the global criterion K_1 : $w(k_{11})=0,239458$, $w(k_{12}) = 0,239458$, $w(k_{13}) = 0,432749$, $w(k_{14}) = 0,088335$
- The WCRI values of the local criteria for the global criterion K_2 : $w(k_{21}) = 0,68334$, $w(k_{22}) = 0,19981$, $w(k_{23}) = 0,11685$
- The WCRI values of the local criteria for the global criterion K_3 : $w(k_{31}) = 0,332516$, $w(k_{32}) = 0,527836$, $w(k_{33}) = 0,139648$
- The summarized WCRI values for the global criterion K_i : $K_1 = 0,527836$, $K_2 = 0,332516$, $K_3 = 0,139648$

And finally, according to *Step 5* of this method (see Section 3.2), and using the multi-criteria ranking formulas (11) - (12), we obtain the following ultimate results of the effectiveness assessment for the considered alternatives (see Table 5), namely

$$X_1 = 0.537, X_2 = 0.671, X_3 = 0.578, X_4 = 0.727 \tag{17}$$

To confirm the reliability of the results given in (17), the comparative analysis with some "best practices" in ITSMS implementation was carried out, using the data of IDC-company [28]. In particular, IDC has reviewed approx. 600 organizations worldwide, which used ITSM for over a year, and in this study especially the prioritization issues of different ITSM-modules implementation were analyzed. In Fig. 5 the result of the performed comparison is shown.

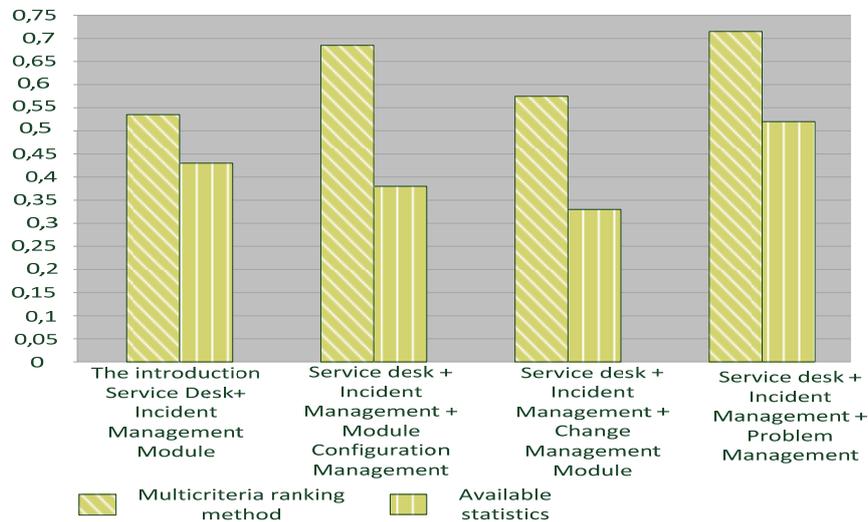


Fig. 5. Graphical representation of the obtained results

As we can see, to provide *Change Management* and *Configuration Management* is necessary to have within an IT-infrastructure database (DB) of IT-configurations, and DB of problem situations as well, these facilities are rather costly for the University, and therefore the implementation of these modules is not a priority task. The most effective ITSM-modules configuration for NTU "KhPI" includes an *Incident Management* module and a *Service Desk* subsystem.

7 Conclusions and Future Work

In this paper we have presented the intelligent approach to increase efficiency of ITSMS, which has to resolve 3 interconnected tasks for its effective usage in a target organization: 1) providing an effective configuration of ITSM-modules according to its specific features and needs; 2) elaboration an integration framework for a given ITSMS with existing EA; 3) advanced incidents management in ITSMS. To solve these tasks in a comprehensive way the interdisciplinary framework is elaborated, which includes: the expert method for multi-criteria ranking of alternative ITSM-modules configurations, the ontological specifications for ITSMS-EA integration, and the approach to enhanced incident management based on the combination of adaptive ontologies and CBR-methodology. To implement the first part of this approach the appropriate software tool was elaborated, and its applicability was tested successfully within the case-study at the NTU "Kharkov Polytechnic Institute".

In future we are going to implement and to test the appropriate software solutions for other tasks in the proposed framework, using such technologies as OWL, BPMN, XML /XLST, and Web-services.

References

1. Office of Government Commerce: ITIL Library. London (2003)
2. International Organization for Standardization. ISO/IEC 20000-1,2: Information Technology-Service Management, Part 1, 2. Geneva, Switzerland: ISO/IEC (2005)
3. Braun, C., Winter, R.: Integration of IT Service Management into Enterprise Architecture. In: Proceeding of SAC'07, Seoul, Korea (2007)
4. ITSM Frameworks and Processes and their Relationship to EA Frameworks. In: A White Paper by: R. Radhakrishnan, IBM Global Technology Services (2008)
5. Valiente, M.-C., Vicente-Chicote, C., Rodriguez, D.: An Ontology-based and Model-driven Approach for Designing IT Service Management Systems. In: Int. Journal of Service Science, Management, Eng. and Techn., 2(2), pp. 65--81 (2011)
6. Valiente, M.-C., Barriocanal-Garcia E., Sicilia, M.-A.: Applying an Ontology Approach to IT Service Management for Business-IT Integration. In: Knowledge-Based Systems, vol., 28, pp. 76--87 (2012)
7. Official Web-site of the Protocol, Ltd. company, <http://www.protocolsoftware.com/hp-openview.php>
8. Official Web-site of the BMC Software company, <http://www.bmc.com/products/remedy-itsm/solutions-capabilities/it-service-management-suite.html>
9. Official Web-site of the OTRS Group company, <http://www.otrs.com>

10. Official Web-site of the SourceForge code repository, <http://sourceforge.net>
11. Tkachuk M. V., Sokol V.Y.: Some Problems on IT-infrastructure Management in Enterprises: State-of-the-Art and Development Perspective. *East-European Journal on Advanced Technologies*, 48 (6/2), 68–72 (2010) (in Russian)
12. Official Web-site of the Cleverics company, <http://www.cleverics.ru/en>
13. Saaty, T. L.: *Fundamentals of the Analytic Hierarchy Process*. RWS (2000).
14. Jabrailova Z. Q.: A Method of Multi-Criteria Ranging for Personnel Management Problem Solution. In: *Artificial Intelligent*, 56 (4), pp.130–137 (2009) (in Russian)
15. Valiente, M.-C., Barriocanal-Garcia E., Sicilia, M.-A.: Applying Ontology-Based Models for Supporting Integrated Software Development and IT Service Management Processes. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 42(1), 61–74 (2012)
16. Lopez-Fernandez, H., Fdez-Riverola, L., Reboiro-Jato, M.: Using CBR as Design Methodology for Developing Adaptable Decision Support Systems. University of Vigo, Spain, pp. 123–145 (2011)
17. Prentzas, J., Hatzilygeroudis, I.: Combinations of Case-Based Reasoning with Other Intelligent Methods. *Int. J. of Hybrid Intelligent Systems*, 55–58 (2009)
18. Rodriguez-Garcia, D., Barriocanal, E., Alonso, S., Nuzzi, C.: Defining Software Process Model Constraints with Rules Using OWL and SWRL. *J. of Soft, Eng., Knowl. Eng.*, 20(4), 533–548 (2010)
19. Prieto, A. E., Lozano-Tello, A.: Use of Ontologies as Representation Support of Workflows. *J. Network and Systems Management*, 17(3), 309–325 (2009)
20. Kang, D., Lee, J., Choi, S., Kim, K.: An Ontology-based Enterprise Architecture. *J. Expert Systems with Applications*, 37(2), 1456–1464 (2010)
21. Pansa, I., Reichle, M., Leist, C., Abeck, S.: A Domain Ontology for Designing Management Services. In: *Proc. 3d Int. Conf. on Advanced Service Computing*, pp. 11–18 (2011)
22. Litvin V.: Multi-Agent Decision Support Systems Based on Precedents that Use of Adaptive Ontology. *Artificial Intelligent*, 54(2) 24–33 (2009) (in Ukrainian)
23. Chung, H.-S., Kim, J.-M.: Learning Ontology Design for Supporting Adaptive Learning in e-Learning Environment. In: *IPCSIT-2012*, vol. 27, Singapore, pp.148–152 (2012)
24. Suh, H., Lee, J.: *Ontology-Based Case-Based Reasoning for Engineering Design*. Design Research Group Manufacturing Engineering Lab (2008)
25. Boursas, L.: Efficient Technical and Organizational Measures for Privacy-Aware Campus Identity Management and Service Integration. In: *Proc. EUNIS'06*, Tartu, Estonia (2006)
26. Knittl, S., Hommel, W.: *SERVUS@TUM: User-Centric Service Support and Privacy Management*. In: J.-F. Desnos., Y. Epelboin (eds.) *EUNIS'07*, Grenoble, France (2007)
27. Brooks, P.: *Metrics for IT-service Management*. Van Haren Publishing (2006)
28. Official Web-site of the International Data Corporation (IDC), <http://www.idc.com>

Refining an Ontology by Learning Stakeholder Votes from their Texts

Olga Tatarintseva^{1,2} and Vadim Ermolayev¹

¹ Department of IT, Zaporozhye National University,
66 Zhukovskogo st., 69063, Zaporozhye, Ukraine

tatarintseva@znu.edu.ua, vadim@ermolayev.com

² Satelliz, 158 Lenina st., P.O. Box 317, 69057, Zaporozhye, Ukraine

Abstract. This paper reports on our experiments evaluating the improvement of OntoElect approach to ontology refinement in the case study with the ICTERI Scope Ontology. OntoElect is based on collecting and assessing the commitment of domain knowledge stakeholders for ontological refinement offerings. We report the improvement with respect to the previous results. Our first experiment evaluates the change in the quality of ontology due to the involvement of domain knowledge stakeholders in semantic annotation of their papers, compared to the previous study in which the annotations were done by knowledge engineers. Our second experiment checks if the result became better after the introduction of the automated term extraction from the full texts of ICTERI papers. Extracted terms are compared to the manual annotations. The results of the experiments verify the proposed ontology changes and are further used for the ICTERI Scope ontology refinement.

Keywords. ICTERI Scope ontology, ontology engineering, domain knowledge stakeholder, term mining, evaluation, refinement

Key terms. KnowledgeEngineeringMethodology, SubjectExpert, SubjectDomain, Metric, Ontology

1 Introduction

Maintaining an ontology in its lifecycle that fits all the requirements of the subject domain stakeholders is a complicated task in ontology engineering which does not have a complete solution so far. The problem is to a large extent in devising a methodology for ontology refinement that enables a complete and timely account for those requirements and maps them to the updated revision of the ontology. One complication is that the stakeholders who own the requirements need to be committed to provide their inputs for ontology refinement. Furthermore, those inputs need to be meas-

ured and applied correspondingly to the utility of their contribution and in a harmonized way to ensure the consistency and validity of result.

This paper reports on the improvement of our OntoElect approach for iterative ontology refinement [1, 2]. The approach has been proposed in [1] using the allusion of elections in which different “ontology offerings” compete for the commitment of the pool of the relevant domain knowledge stakeholders being the “electorate”. OntoElect has been basically validated in an experiment reported in [2] where the approach was detailed by offering voting metrics for the ICTERI ontology built and refined iteratively based on semantic annotation of the pool of papers of ICTERI 2011 conference.

The results of our previous experiment suggested several important technical aspects [2] for improving OntoElect methodology as a whole and the accuracy of our measurements in particular. Some of those aspects have been addressed in the work reported in this paper.

Firstly, our previous experiment was based substantially on the manual annotation of papers. A knowledge engineer assigned key terms or suggested missing terms based on her personal interpretation of the abstract of a paper. By that we mimicked voting by paper authors while keeping them free of extra annotation effort. The lowlights of this approach to annotation were that:

- Domain knowledge stakeholders (paper authors) were in fact not involved in the workflow and therefore not motivated to be committed to the resulting ontology refinement
- The quality of semantic annotations we obtained has been perceived as fairly low because (a) done by a knowledge engineer who is not a subject expert with respect to the annotated paper; (b) the source for this work was just an abstract, not a paper, and its meaning has been interpreted by a knowledge engineer.

To overcome those shortcomings we first decided to involve the authors more actively by requesting that they themselves semantically annotate their submissions to ICTERI 2012¹. It has also been considered as promising to refine the approach by automated extraction of terms from the papers authored by our subject domain knowledge stakeholders. Here we present the results of our experiments which checked how these two refinements helped improving the quality and adequacy of the ICTERI Scope ontology.

Further, the document corpus used in the previous experiment was fairly small in size for assuring reliable judgements about the opinion of the stakeholder community. For improving on that we continued the collection of ICTERI papers which has been extended by all papers of ICTERI 2012.

We first repeat the previous experiment [2] based however on the document corpus of ICTERI 2012 papers semantically annotated by their authors. We then focus on answering the question about annotation quality by: (i) performing automated term extraction from the full texts of our complete document corpus (ICTERI 2011 and 2012); and (ii) comparing the results of automated term extraction to the outputs of manual semantic annotation.

¹ See <http://isrg.kit.znu.edu.ua/icteriwiki/index.php/ICTERI-Terms>

The remainder of the paper is structured as follows. Section 2 briefly reviews the related work in relevant fields. Section 3 outlines the OntoElect approach to ontology refinement and presents the case study dealing with ICTERI Scope Ontology as well as the document corpus at our disposal. Section 4 sets up our experiments by describing the workflow, evaluation metrics, and used tools. Section 5 presents and discusses the results of our experiments. The paper is further concluded and our plans for the future work are outlined.

2 Related Work

One of the possible ways to check if a conceptualization of a domain is correct and complete is to evaluate the model against the interpretation of the meaning of the representative set of relevant documents. The document corpus will be relevant and representative if it covers the majority of the views by the domain knowledge stakeholders. Their interpretations may be collected and further analysed for refining the ontology using different techniques which may be sought in several areas of research and development. In this section we briefly outline the related work in the relevant fields of research and refer to our previous publication [1] for a more in-depth and detailed coverage.

One of the popular relevant research areas studying how interpretations are collected is collaborative or social tagging and annotation. A good survey of the field is [3] where the use of tags for different purposes and associated shortcomings are analysed. Semantic annotation and tagging approaches further refine social tagging techniques by offering the collections of terms that are taken from taxonomies, folksonomies, or thesauri [4]. Hybrid approaches for collaborative tagging and annotation aiming at the enrichment of seed knowledge representations by a user community are reported for example in [5].

One of the promising approaches focused, besides collecting interpretations or subjective conceptualizations, on motivating more people to take part in developing or refining ontologies is offering a game with a purpose to intended users. Following this approach, ontology development or refinement can be implicitly embedded in a game software. There ontology elements are created, updated, and validated implicitly in the background [6]. Gaming approach has also been tried for evaluating how well ontological specifications fit to the interpretations of random users (FACTory Game by Cycorp, <http://game.cyc.com/>). Several game scenarios have been developed [5] for ontology building and refinement, ontology matching, annotating content using lightweight ontologies. Those are similar to our OntoElect approach. Both approaches offer possibilities to identify whenever users start to agree on and share commitment to certain ontological items.

Social and gaming approaches that involve the direct participation of human stakeholders are complemented by the plethora of research results in automated knowledge extraction or ontology learning. This strand of research involves the stakeholders indirectly – through making use of their professional outputs, like authored texts. A comprehensive survey of the techniques used to learn ontologies from texts is [7]. In

the second experiment we present in this paper only term extraction using the TerMine tool [8] has been performed.

Yet one more important aspect in developing or refining an ontology is the re-use of the other ontologies or their most relevant parts to the developed ontology. In this context the ontology meaning summarization approach [#] makes good sense for helping an ontology engineer choose the most relevant and valuable parts for re-use. The approach is based on detecting the “key concepts” of an ontology under analysis which best characterize its meaning. The key concepts are determined using a combination of criteria from lexical statistics, taxonomy graph analysis, and popularity based on a number of hits. Especially in using the popularity and coverage metrics, this approach coincides well with our approach (OntoElect). OntoElect is however used not for summarizing but refining an ontology based, among other things, on assessing the coverage and popularity of the Key Terms. Besides that the mechanisms of obtaining the measures are different. From the other hand, OntoElect does not yet consider ontology re-use as one of important mechanisms for refinement. Hence, combining some features of [9] in OntoElect may be enriching.

3 ICTERI Case Study

The idea of OntoElect approach [1] was inspired by public election campaigns. Just as the leader in a public election campaign gets the major part of the electorate’s commitment to win, the extent of the domain knowledge stakeholders’ commitment hints about the quality and completeness of the ontology. Following this allusion, the votes of the domain knowledge stakeholders for alternative ontology offerings are collected and used as the measure of their commitment. The ontology offering that collects the biggest share of votes could therefore be considered as the best and most complete.

In our case study the OntoElect approach is applied for refining the ICTERI Scope Ontology in the iterative ontology engineering experiment. Our domain knowledge stakeholders are the authors of ICTERI papers. Ontology offerings in the reported work are the structural contexts² in the five thematic areas of the ICTERI scope offered to the authors for choosing the appropriate key terms to annotate their papers.

As this data had to be selected we decided to simulate the opinions of the electorate by annotating the papers of ICTERI 2011 manually. For this we extracted the terms which were specified as the list of ICTERI Key Terms if it was possible. In some cases we had to add Missing Concepts (also called Missing Key Terms) for the papers, if such terms did not exist in the list.

So, we received three semantic annotation types:

- KeyWord – for the key words, which were selected by the authors
- KeyTerm – for the terms which were selected manually and were found in the list of the ICTERI terms

² A structural context, as suggested e.g. in [10], is composed of a central concept with all his domain and object properties and the concepts connected to the central concept by these object properties.

- MissingConcept - for the terms, which did not exist in the list of the ICTERI terms, but were covered during annotation

One use of the particular term was considered to be one vote for the selected term. The votes were normalized as frequencies of use. Such information allowed us to measure the popularity of each semantic context, circumscribe the most frequently demanded part of the ontology and make suggestion about the completeness of the ontological offerings.

For the papers of ICTERI 2012 we requested that the authors annotate their papers not only using the freely chosen key words, but also using the terms found in ICTERI scope ontology. As the result the corpus for the further analysis was increased. The data provided by the authors can be accepted as more authentic than that which was obtained ourselves, as they are the real domain experts for the field they study. The analysis of the received data is presented in Section 5.

But even when we use the information presented by the authors, and the results of our manual annotation we can't guarantee that this information is accurate enough for applying it to the ontology refinement process. To obtain the experiment we needed to have results received in different ways because we wanted to achieve the impartial assessment of OntoElect approach. Before applying the results in ontology refining process we decided to check them with freely available tool for text mining.

For our experiment we chose one of the services provided by the National Centre for Text Mining (NaCTeM). As reported in the official website of NaCTeM³ it is the first publicly-funded text mining centre in the world. It provides text mining services in response to the requirements of the UK academic community. NaCTeM is operated by the University of Manchester.

4 Experimental Set-up and Tools

To control the results of the experiment we have to understand which main questions we are going to answer after its realization and how to measure these results.

Our measurable objectives for the experiment have been formulated as follows [2]:

- Does the ontology fit to the requirements of the subject experts in the domain?

The fitness of the ontological offering will be measured as a ratio of the average frequency of use of the available Key Terms (positive votes) to the similar for the missing Key Terms (negative votes). Special attention will be paid to the freely chosen key words that are identical to the available Key Terms. Those will be considered as extra positive votes for the semantic context of the Key Term.

- Is there a particular part in the ontology that is the most important for the stakeholders?

The importance of an ontology fragment comprising particular concepts will be measured as frequency of use of these concepts (positive votes). Fragments of different importance will also be presented as percentiles.

³ Official web site of National Centre for Text Mining <http://www.nactem.ac.uk/>

- Is there a part in the ontology that could be dropped as the stakeholders do not really require it?

Similarly to importance, these ontology fragments will be outlined using low frequency of use percentiles.

- What would be a most valuable addition to the ontology that will substantially improve stakeholders’ commitment to it?

The papers have been annotated using missing Key Terms and freely chosen keywords. Those missing Key Terms that are frequently used will form the core of this effective extension. If some of the keywords are also used frequently by the authors they may become good candidates for the inclusion in the effective extension as well. Special attention will be paid to the freely chosen key words that are identical to the missing Key Terms. Those will reinforce the votes on the addition to the ontology.

The flow of activities has been organized in three consecutive phases as presented in Fig. 1. The description of each phase in details is presented in [2].

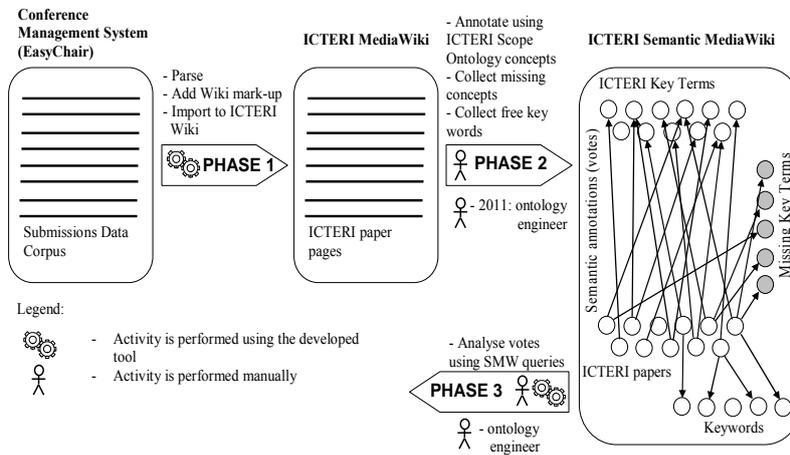


Fig. 1. The workflow for processing ICTERI papers and collecting stakeholders’ votes repeated according to [2]

At phase 1 we have extracted the semi-structured information about the papers accepted for ICTERI 2012 and transformed these into the collection of paper articles in the ICTERI Wiki. At Phase 2 we extracted the freely chosen KeyWords and the KeyTerms from the ICTERI Scope ontology assigned by the authors and added these to the semantic annotations of the papers. In several cases we detected considerable meaning gaps between the extracted key words and Key Terms when annotated the papers manually. Therefore, we opted to add the missing Key Terms to the corresponding semantic annotations. As a result of this Phase the semantic relationships between the pages of `Category:Paper` and the pages of `Category:Concept` have been specified as semantic properties. These semantic properties allowed us to receive all the measurements planned for the evaluation experiment. These measurements have been done using different Semantic MediaWiki queries at Phase 3.

Compared to the previous year experiment [2], we automated the extraction of the frequency of use statistics which made the process less error prone and faster. For that the SMWAskAPI⁴ extension of the Semantic MediaWiki has been used. This extension supports semantic queries of #ask and enables the use of the corresponding API for executing Semantic MediaWiki ask queries.

Each page of the ICTERI Wiki uses semantic tagging. An example of the Semantic properties specified for pages in `Category:Paper` is given in Fig. 2.

For our analysis we used the pages from `Category:Paper` and `Category:Workshop` with the property `hasPublicationYear` equal to 2012, namely the values of the semantic properties `hasKeyWord`, `hasKeyTem`, and `MissingConcept`.

Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering	
HasAuthor	Olga Tatarintseva + ⓘ , Vadim Ermolayev + ⓘ , Anna Fensel + ⓘ
HasKeyTerm	KnowledgeEngineeringMethodology + ⓘ , SubjectExpert + ⓘ , Collaboration + ⓘ , Approach + ⓘ
HasKeyWord	Ontology + ⓘ , Stakeholder commitment + ⓘ , Collaboration + ⓘ , Ontology engineering + ⓘ , Ontology election + ⓘ
HasLanguage	English + ⓘ
HasPresentation	Presentation: Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering + ⓘ
HasPublicationYear	2011 + ⓘ
HasTitle	Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering + ⓘ
MissingConcept	ConceptualModeling + ⓘ
Modification date	31 March 2012 18:59:28 + ⓘ
PublicationURL	Http://oeur-ws.org/Vol-716/ICTERI-2011-CEUR-WS-paper-4-p-65-81.pdf + ⓘ
Categories	Paper
<small>hide properties that link here</small>	
Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering + ⓘ , Presentation: Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering + ⓘ	HasTitle
Burden-or-Gem + ⓘ	redirect page

Fig. 2. Semantic properties for the ICTERI Wiki page in the `Category:Paper`

The scripts for analyzing these values were coded in Python. Some steps were also implemented using shell scripting. As outputs we have received:

- The list of KeyWords, KeyTerms, and MissingConcepts for each article, if they were defined
- The overall number of the papers according to the values of the properties `hasPublicationYear`, and the selected `Category`
- The number occurrences of each KeyWord, KeyTerm and MissingConcept.

The analysis and discussion of the results is given in Section 5.

To perform our second experiment we applied the Term Management System named TerMine, which identifies key phrases in text. It uses C-value [8], a domain-independent method for automatic term recognition (ATR) which combines linguistic and statistical analyses with the emphasis on the statistical part. The linguistic analysis enumerates all candidate terms in a given text by applying part-of-speech tagging, extracting word sequences based on adjectives/nouns, and stop-list. The statistical

⁴ See the description of the SMWAskAPI on <http://www.mediawiki.org/wiki/Extension:SMWAskAPI>

analysis assigns a candidate term to a termhood by using the following four characteristics:

- The occurrence frequency of the candidate term
- The frequency of the candidate term as part of other longer candidate terms
- The number of these longer candidate terms
- The length of the candidate term

The data corpus for this term extraction and analysis was the merge of the pools of ICTERI 2011⁵ and ICTERI 2012⁶ papers published in the respective proceedings, and consisted of 63 papers. The papers from both proceedings volumes have been merged in a single file and uploaded for processing by TerMine. The workflow for the second experiment is pictured in Fig. 3.

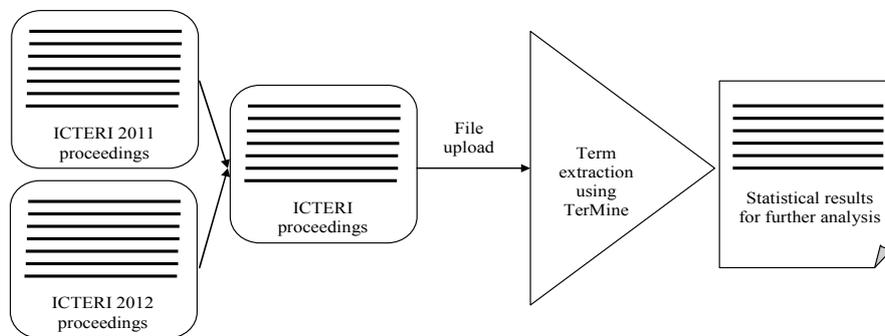


Fig. 3. The workflow for conducting the second experiment for term mining and analysis

The data processed in the pipeline is illustrated by the example of a single paper [1] in Fig. 4.

The results of term mining were provided in several forms. All the terms defined in the text were highlighted by colour markings (upper part of Fig. 4a). The information about the overall number of the terms mined from the text was also given (433 terms listed – in the bottom of Fig. 4a). The terms were also presented in the table view, each preceded with the assigned rank number and followed by the statistical score measure (lower part of Fig. 4a). The rank of a term means the position of each term in the table sorted by the score; the rank values of the terms with the same score are equal. The scores were computed automatically using the Term Recognition technique [8] which uses the information about the frequencies of term occurrence. This approach is essentially a shallow bag of terms extraction technique – therefore the output needs to be post-processed as described using our single paper data example. For this example the number of extracted terms was 433 which is obviously too many. To compare, the authors were advised to assign 3-5 KeyTerms to their papers which best describe its meaning. Among those extracted terms that we needed to sort out were also names, affiliations, cities, etc, which had no semantic relationship to the

⁵ <http://ceur-ws.org/Vol-716/>

⁶ <http://ceur-ws.org/Vol-848/>

meaning of the paper. Also, it has been assumed that the terms with a low number of occurrences in text have a negligent semantic contribution and may also be filtered out – so only the higher ranked part of the term list may be considered.

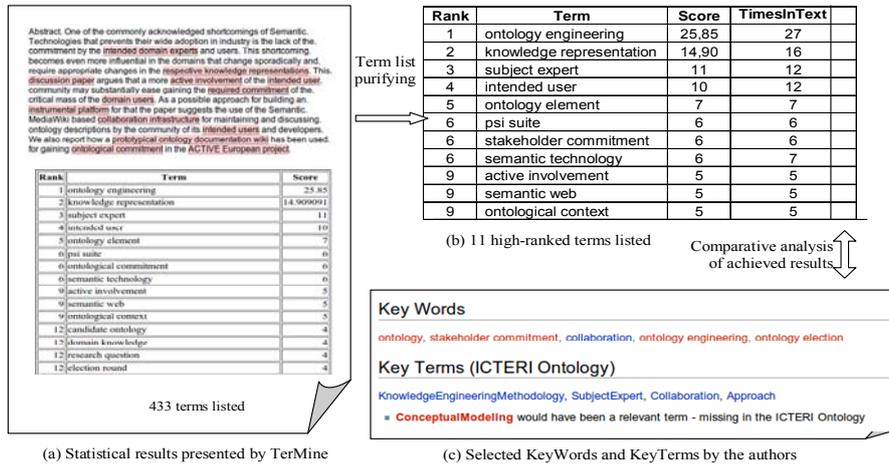


Fig. 4. An example of the data processed in the term mining experiment

While post-processing the list of mined terms we decided to leave only the terms, which were used more than 5 times, and had score more than 5 points. Applying this threshold returned 11 of 433 terms for the example outlined in Fig. 4., which constitutes only 2.54 per cent of the overall number of the mined terms. The manual check of the example however indicates that these 11 high ranked terms indeed contribute most significantly to describing the semantics of the corresponding paper (Fig. 4b).

The right part of Fig. 4 allows to compare the result of term extraction (Fig. 4b) with the output of manual semantic annotation (Fig. 4c) for the selected example paper. A mechanical comparison reveals substantial difference, which however is not that big after manual mapping of the extracted terms to the concepts of the ICTERI Scope ontology. In fact there is a subset of extracted terms that could be directly mapped into the Key Terms of the ontology: subject expert; ontology engineering (as a methodology). Another group is relevant to the assigned KeyWords: ontology, stakeholder commitment, ontology engineering. Some are synonymic in the context of this paper: stakeholder and intended user. Some represent the meaning which is too fine-grained for a semantic annotation: ontology element. And, which is most important, some are the new valid candidates for the inclusion into the ICTERI Scope ontology: knowledge representation, semantic technology, semantic web.

5 Results and Discussion

In this section we present the results of the experiment. The set up of all its stages is described in Section 4. The discussion of the experiment results is structured along the measurable items.

The frequency of use diagrams (Fig. 5 and 6) are built in regard to the total amount of the papers and the number of occurrences of a particular term.

Similar work was reported in our previous publication [2]. In it we described the mechanism of ranging. The actual experiment is based on the previous results. But they changed as the document corpus we are working with has increased and the data to work with has changed. As we consider OntoElect approach in the case study of iterative refinement of the ICTERI Scope Ontology such changes are greatly important. The diagrams which show these changes are shown below:

- For the KeyWords which were selected by the authors manually (only those, which were chosen by at least two authors, Fig. 5)
- For the KeyTerms, which were selected from the ICTERI ontology terms (Fig. 6)

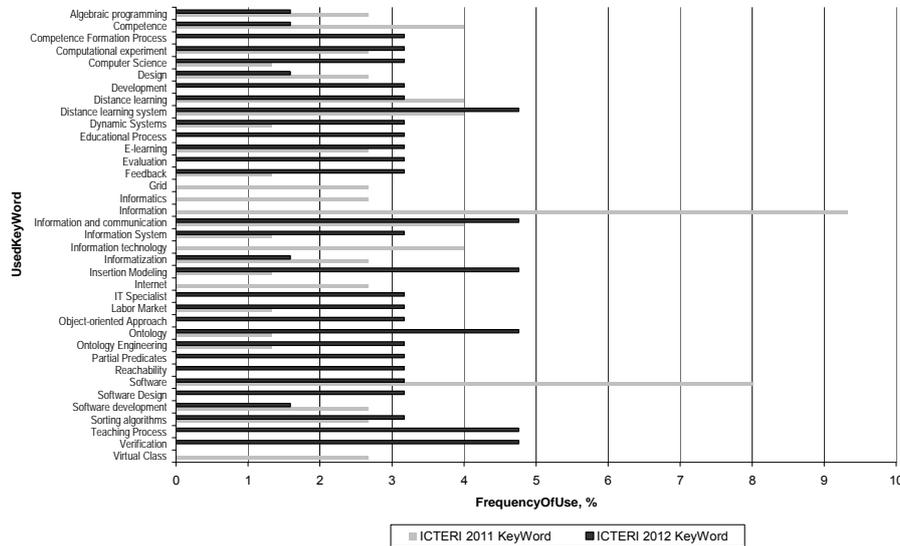


Fig. 5. The frequency of use of the freely chosen KeyWords

We did not provide a frequency of use diagram comparing the Missing Key Terms because the difference in the results of 2011 and 2012 is tiny and could be neglected. We provided the comparison analysis for KeyWords and Missing Key Terms lists (Fig. 7). To compute the range of use of each term we divided each frequency of use index by the frequency value of the most popular term, which range of use was taken as 100 per cent. These terms are not the part of the ontology, but are the most possible candidates.

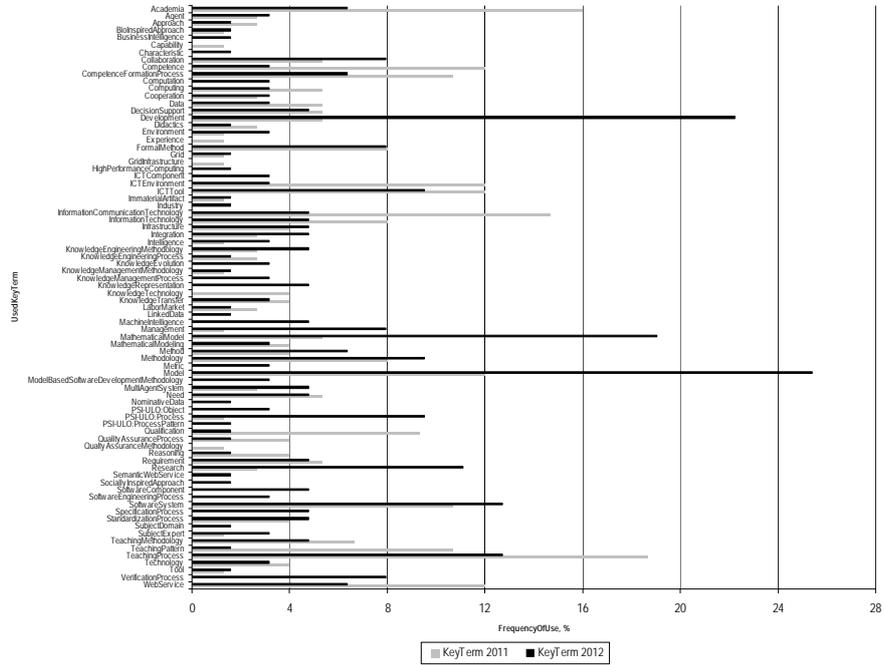


Fig. 6. The frequency of use of available KeyTerms

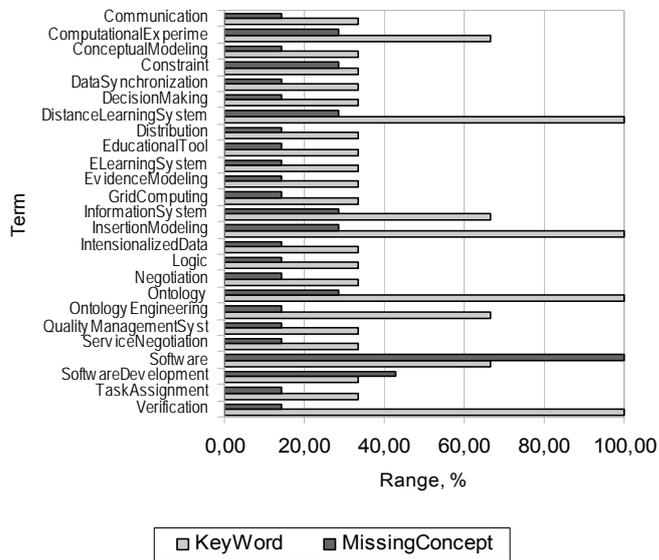


Fig. 7. The range of use for Missing Key Terms and KeyWords

The results which we received at the previous step can be merged and we can receive the new version of the potential ontology offering. But before doing this we will look into the results of the text mining experiment.

Using TerMine tool for automation of the knowledge mining process we received some interesting results. The overall number of the found terms was 8487. All of the selected terms were graduated and received the position in the rank table. Studying the results it is obvious that the number of the terms selected by the data mining tool is too big.

It was decided to leave in the rank table only those terms which have the score of 10 and more. The popularity of the terms which were picked up is evident. The total number of such terms is 157. It is easy to count that it makes up less than 2% of all the terms proposed by the tool. After refining the list and deletion of the superfluous information only 140 terms left.

To compare the frequencies of use provided by the TerMine and our own calculations we decided to use percentage method (similar to that used for building the diagram in Fig. 7). We took the maximal value for each group of concepts as 100 per cent and divided it by the frequency of use value of a particular term. As a result each term got the value, called the range, which could be compared with the ranges of the other terms. We analyzed the three groups of mined term matches to the: (i) KeyWords; (ii) MissingConcepts; and (iii) KeyTerms. As the number of the Missing Concepts is not too big we decided to combine them with the Key Words in the diagram (Fig. 8).

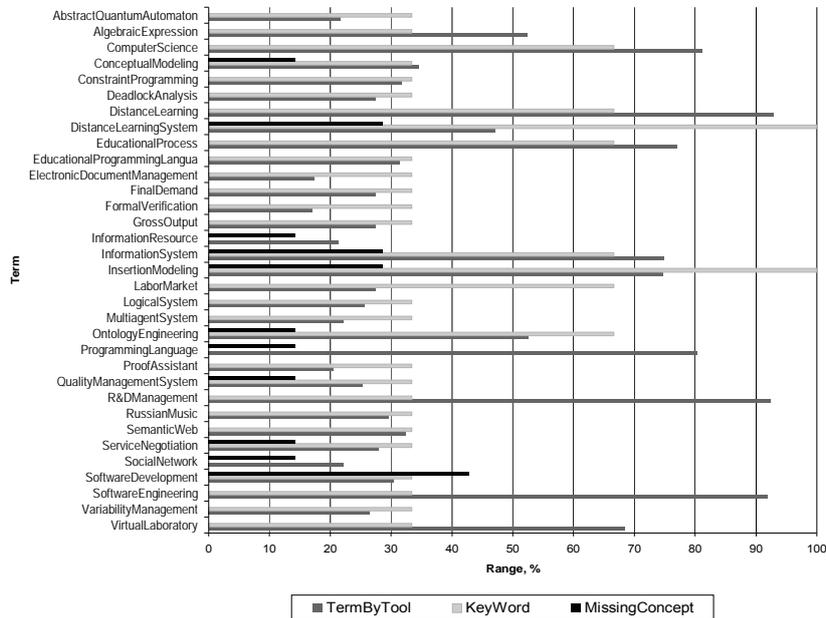


Fig. 8. The range of use for terms detected by tool, Key Words and Missing Concepts. Range values were normalized by the frequency of use of the highest scored extracted term (100)

After the automated search of the identical terms in the lists of KeyTerms and terms mined by tool we discovered that some of them were missed as the search did not use the rules of common sense and the relations described in the ontology. For example, according to the ICTERI Scope ontology the term Integration subsumes to PSI-ULO:Process. Knowing this fact we understand that the term IntegrationProcess is just the same as the term Integration. But this match is not obvious for the simple search and will not be detected.

Therefore, to find the matches in the lists of KeyTerms and terms mined by the tool we decided to undertake a more careful analysis. We scanned the list of the KeyTerms for matching the ToolTerms manually. Besides for this process we used the whole pool of 8487 terms mined by the tool. The result is pictured in the diagram (Fig. 9).

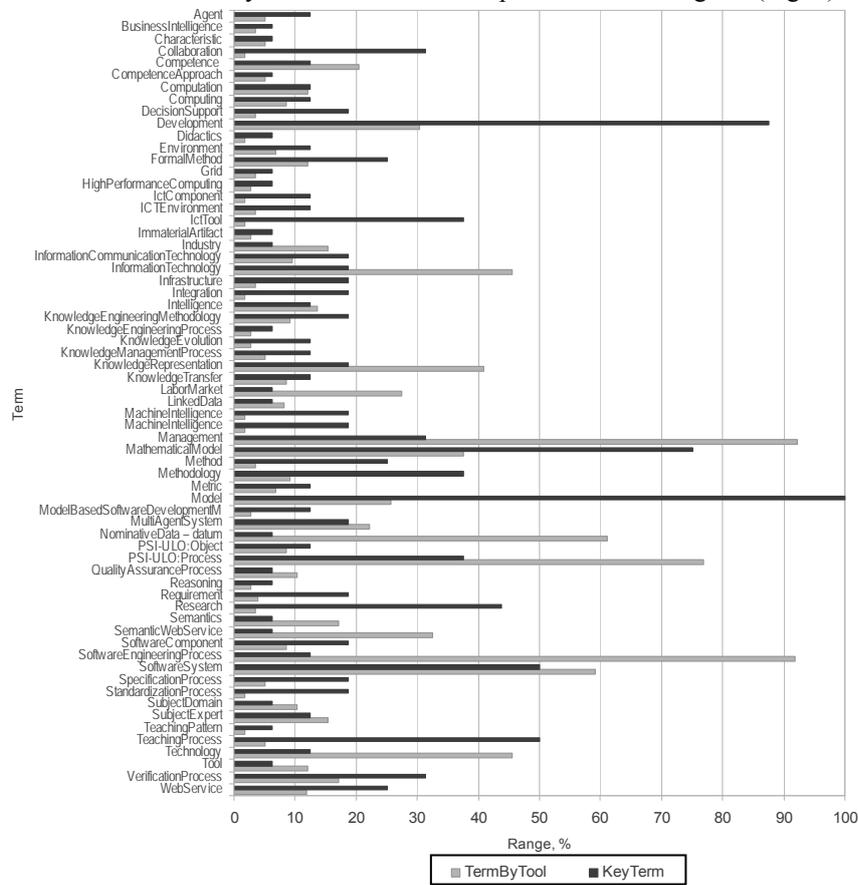


Fig. 9. The range of use for the KeyTerms and the terms extracted by the TerMine tool

6 Concluding Remarks and Future Work

The paper reported on the experiment evaluating the improvement of OntoElect approach to ontology engineering in the case study with the ICTERI Scope Ontology. In particular, the approach has been used to evaluate the validity of the papers' annotation by drawing knowledge stakeholders to their own papers' annotation and by studying their quality using voting for full texts.

The experiment consisted of two stages. The first one was based on the comparison analysis of the papers presented during international conferences ICTERI 2011 and ICTERI 2012. The second one was dedicated to performing automated term extraction from the full texts and comparing the results of automated term extraction to the outputs of manual semantic annotation.

Achieved results stress the important parts of the ontology and those which are less popular among the authors. The comparison analysis of the first experiment shows how the situation changed during two years. The KeyWords and MissingConcepts which have high frequency of use values, especially if they are named in both lists, are the first candidates to become the new part of the ontology.

The second experiment shows which ontological offerings agree with the terms mined by tool and which numeric characteristics these matches have. The terms extracted by the tool and their matches with the KeyWords and MissingConcepts, which have range more than 50 per cent, are also good candidates to be added to the ontology. Besides, the degree to which the extracted terms match the KeyWords and KeyTerms indicate about the adequacy of paper annotation. Overall the overlap between the meanings of the extracted terms and the KeyTerms measures the range of so to say the similarity in the meanings of the papers within the corpus and the ontological offerings aimed at covering these meanings. The quantitative results of our experiments still need to be processed and analyzed more thoroughly before deciding about the implementation of the changes to the ontology. Besides that several other aspects still need to be researched in our future work.

Firstly, the document corpus used in the case study, though growing, is still not very big to allow robustly applying the majority of traditional knowledge extraction techniques. At the moment it could only be stated that the information we have now is enough to prove the concept – i.e. the validity of the approach based on the assessment of and account for domain knowledge stakeholder opinions, implicitly reflecting their needs. After applying the refinements suggested by the stakeholders, the ontology still needs to be evaluated and validated using other methods.

Secondly, in this paper we reported about only a partial and shallow way of extracting knowledge from paper texts. A possible refinement to this preliminary solution could be sought in using a hybrid iterative knowledge extraction workflow that incrementally adds ontology elements to the “ontology learning layer cake (c.f. [11])”.

References

1. Tatarintseva, O., Ermolayev, V., Fensel, A.: Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering. In: Ermolayev, V. et al. (eds.) Proc. 7-th Int. Conf. ICTERI 2011, Kherson, Ukraine, May 4-7, 2011, CEUR-WS.org, vol-716, ISSN 1613-0073, 65–81, online (2011)
2. Tatarintseva, O., Borue, Yu., and Ermolayev, V.: OntoElect Approach for Iterative Ontology Refinement: a Case Study with ICTERI Scope Ontology. In: Ermolayev, V. et al. (eds.) Proc. 8-th Int. Conf. ICTERI 2012, Kherson, Ukraine, June 6-10, 2012, CEUR-WS.org, vol-848, ISSN 1613-0073, 244, online (2011)
3. Gupta, M., Li, R., Yin, Z., Han, J.: Survey on Social Tagging Techniques. SIGKDD Explorations 12(1), 58–72 (2010)
4. Uren, V., Cimiano, P., Iria, J., Handschuh, S., Vargas-Vera, M., Motta, E., Ciravegna, F.: Semantic annotation for knowledge management: Requirements and a survey of the state of the art. *Science. Services and Agents on the World Wide Web* 4(1), 14–28 (2006)
5. Hunter, J., Khan, I., Gewrber, A.: HarvANA – Harvesting Community Tags to Enrich Collection Metadata. In: Paepcke A, Borbiha J, Naaman M (eds.) 8th ACM/IEEE-CS Joint Conference on Digital Libraries, 147–156. ACM New York, New York (2008)
6. Siorpaes, K., Hepp, M.: Games with a Purpose for the Semantic Web. *IEEE Intelligent Systems* 23(3), 50–60 (2008)
7. Wong, W., Liu, W., and Bennamoun, M.: Ontology learning from text: A look back and into the future. *ACM Comput. Surv.*, 44(4), Article 20, 36 pages. DOI=10.1145/2333112.2333115 (September 2012)
8. Frantzi, K., Ananiadou, S. and Mima, H.: Automatic recognition of multi-word terms. *Int. J. of Digital Libraries* 3(2), pp.117-132 (2000)
9. Peroni, S., Motta, E., d'Aquin, M. Identifying Key Concepts in an Ontology, through the Integration of Cognitive Principles with Statistical and Topological Measures. In: Proc. 3rd Asian Semantic Web Conference (ASWC 2008), Dec 08-11, 2008, Bangkok, Thailand (2008)
10. Ermolayev, V., Copylov, A., Keberle, N., Jentzsch, E., Matzke, W.-E.. Using Contexts in Ontology Structural Change Analysis.. In: Ermolayev, V., Gomez-Perez, J.-M., Haase, P., Warren, P, (eds.) CIAO 2010, CEUR-WS, vol. 626 (2010)
11. Wong, W., Liu, W., Bennamoun, M.: Ontology Learning from Text: a Look Back and into the Future. *ACM Comput. Surv.*, 44(4), Article 20, 36 p., <http://doi.acm.org/10.1145/2333112.2333115> (2012)

Answering Conjunctive Queries over a Temporally-Ordered Finite Sequence of ABoxes Sharing one TBox

Natalya G. Keberle *

Zaporozhye National University, Dept. of Information Technologies
66, Zhukovskogo str. 69063, Zaporozhye, Ukraine

nkeberle@gmail.com

Abstract. Ontology-based data access (OBDA) assumes that data in a database are mediated with a conceptual layer, available for clients and hiding data storage details. Ontologies are good candidates for such a conceptual layer presentation, whereas databases are good for huge data storage. One of the interesting applications of OBDA is checking a finite set of constraints defined in some language against a temporally-ordered sequence of ABoxes sharing one TBox, where each constraint is considered as a conjunctive query. Presented is one algorithm of conjunctive query answering for such a language, proved are its termination, soundness and completeness.

Keywords. Ontology-based data access, temporal conjunctive query language, description logic knowledge base

Key terms. KnowledgeEvolution, KnowledgeManagementProcess, DecisionSupport

1 Introduction

Ontology-based data access (OBDA) [1] assumes that data in a database are mediated with a conceptual layer, available for clients and hiding data storage details. Ontologies are good candidates for such a conceptual layer presentation, whereas databases are good for huge data storage.

The benefits from combination of databases and knowledge bases are as follows:

- database management is the mature field of research, there is a lot of commercially and freely available DBMSs, showing good-to-excellent performance on large datasets. It is an obvious place to store the assertional part of some knowledge base, i.e. an ABox;
- a TBox often requires a reasoning support to deduce additional assertions, axioms and to check the consistency of a knowledge base.

* The work done during the research visit to Dresden University of Technology, sponsored by The Ministry of Education and Science, Youth and Sport of Ukraine

At the same time, employing such an approach is rather challenging due to significant differences between relational database systems and ontology languages, based on Description Logics, such as OWL. At first, relational databases adopt a closed-world semantics, i.e. all facts that are not explicitly stated to be true are assumed to be false. In contrast, OWL is based on an open world semantics which does not require one to fix the truth value of every fact and is more similar to an incomplete database. Second, relational databases are unaware of the intensional part of a knowledge base (called a TBox).

Research has been done so far in the OBDA field considers only one ABox stored in a data source, that is an actual set of assertions on individuals and their pairs. However, real applications show that ABox is changing over time. The examples of such dynamic systems can be easily found in practice: environmental conditions, air traffic load, computer system load and performance, health control for the people suffering from serious diseases. Therefore, in some applications of situation awareness [2], there is a need to store an archive of ABoxes, keeping ABoxes actual at different time points. Temporal logics are often used as the means to formulate constraints a dynamic system should obey during its work.

The main results of the paper are:

- for the point-based linear finite time structure elaborated is the language of unions of temporal conjunctive queries, which allows to evaluate atemporal unions of conjunctive queries at different time points;
- proposed is an algorithm of answering a union of temporal conjunctive queries, which harnesses set-theoretic operations on atomic queries answer sets. Proved are its termination, soundness and completeness;

The paper is organized as follows: in the next section a language of unions of temporal conjunctive queries is introduced and the definitions of main reasoning tasks available for such a query language are presented. In the section 3 the algorithm of answering temporal unions of conjunctive queries is presented and illustrated in examples. The section 4 is dedicated to the proofs of logical properties of the algorithm. The section 5 discusses the related work in the field of conjunctive queries answering.

2 Conjunctive Queries: Syntax, Semantics

Assume there is a knowledge base $\mathcal{K} = (\mathcal{A}, \mathcal{T})$, where \mathcal{T} is a set of concept axioms (a TBox), \mathcal{A} is a set of assertional axioms (an ABox). Fix a language of a knowledge base to \mathcal{ALC} [3]. An interpretation of \mathcal{K} , named \mathcal{I} , is a pair $(\cdot^{\mathcal{I}}, \Delta)$, where Δ is a domain of individuals, obeying unique name assumption (UNA) and $\cdot^{\mathcal{I}}$ is an interpretation function, which assigns every concept C a set $C^{\mathcal{I}} \subseteq \Delta$, every atomic role R a binary relation $R^{\mathcal{I}} \subseteq \Delta \times \Delta$, and every individual name a an individual $a^{\mathcal{I}} \in \Delta$. Assertional axioms are $C(a)$ - concept assertion and $R(a, b)$ - role assertion.

Query answering is the extension of a well known task of *instance checking*: given a knowledge base \mathcal{K} and an assertion α . Check whether this assertion is entailed by an ABox of \mathcal{K} .

2.1 Conjunctive Queries Basics

Let $\text{Vars}(Q)$ be a set of all distinguished and non-distinguished variables which appear in a query Q , let $\text{Inds}(Q)$ to denote the set of all individual names which appear in query Q and $\text{Terms}(Q)$ to denote the set of all terms in Q , i.e. $\text{Vars}(Q) \cup \text{Inds}(Q)$. Let us formally define conjunctive queries and Boolean conjunctive queries for a well-elaborated language \mathcal{ALC} [3].

Definition 1 (Conjunctive query, Union of conjunctive queries). *Let \mathbf{x} , \mathbf{y} , \mathbf{c} are respectively tuples of distinguished variables (answer variables), of non-distinguished variables and of individual names, and t , t_1 , t_2 are terms in $\text{Terms}(Q)$. A conjunctive query (CQ) is an expression of the form*

$$\text{conj}(\mathbf{x}, \mathbf{c}) = \exists \mathbf{y}. q_1 \wedge \dots \wedge q_m,$$

where

$$q_i ::= C(t) \mid r(t_1, t_2)$$

A Boolean conjunctive query is a CQ without answer variables.

A union of conjunctive queries (UCQ) is a disjunction of conjunctive queries (CQs) of the form

$$Q(\mathbf{x}) = \{\mathbf{x} \mid \text{conj}_1(\mathbf{x}, \mathbf{c}) \vee \dots \vee \text{conj}_n(\mathbf{x}, \mathbf{c})\}$$

Example 1. The example of a query asking about all students that attend some courses and take some exams could be as follows:

$$Q(x) = \{x \mid \exists y. \text{takeCourse}(x, y) \wedge \text{takeExam}(x, y)\}$$

This query can be modified to a Boolean query by substitution of x with an individual name:

$$Q(x) = \{\mid y. \text{takeCourse}(\text{"Eldora"}, y) \wedge \text{takeExam}(\text{"Eldora"}, y)\}$$

We use $|Q|$ to denote the *size* of Q - the number of symbols required to build the query. The *arity* of a query will be the number of answer variables in the query. If all terms in Q are individual names, we say Q is *ground*. We write $Q(\mathbf{c})$ for a query whose answer variables \mathbf{x} are substituted by \mathbf{c} , $Q(\mathbf{x})$ for a conjunctive query and simply Q for a Boolean conjunctive query. Sometimes we write x_1, \dots, x_n instead of \mathbf{x} , and similarly for \mathbf{y} and \mathbf{c} .

Given an \mathcal{ALC} -knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, an interpretation \mathcal{I} satisfies a query $Q(\mathbf{x})$ iff the interpretation function can be extended to the variables in $Q(\mathbf{x})$ in such a way that \mathcal{I} satisfies every term in $Q(\mathbf{x})$. A query $Q(\mathbf{x})$ is true w.r.t. \mathcal{K} (written $\mathcal{K} \models Q$) iff every interpretation that satisfies \mathcal{K} also satisfies Q .

Definition 2 (Query answering, query entailment). *Given a query $Q(\mathbf{x})$ with a tuple of answer variables \mathbf{x} , and a knowledge base \mathcal{K} , a tuple of individuals \mathbf{c} with the same arity of \mathbf{x} is an answer for Q in \mathcal{K} if $\mathcal{I} \models Q(\mathbf{c})$ for every model \mathcal{I} in \mathcal{K} .*

Given a Boolean conjunctive query Q , and a KB \mathcal{K} , query entailment is a task to decide whether $\mathcal{K} \models Q$ if $\mathcal{I} \models Q$ for every model \mathcal{I} of \mathcal{K} .

Given a conjunctive query $Q(\mathbf{x})$, a tuple of individuals \mathbf{a} , and a KB \mathcal{K} , query answering is a task to decide whether \mathbf{a} is an answer for $Q(\mathbf{x})$ in \mathcal{K} .

3 Temporal Conjunctive Queries

Let $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$ be a knowledge base with a sequence of ABoxes sharing one TBox. Let's describe a query language extending the language of conjunctions of positive existential formulae built from query atoms. Having in mind linear temporal logic *LTL* (see e.g. [4]), this language allows for the following temporal operators: \circ (*next*), \circ^- (*previous*), \mathcal{U} (*until*), \mathcal{S} (*since*).

Definition 3. Temporal conjunctive query (TCQ) Ψ is an expression

$$tconj(\mathbf{x}, \mathbf{c}) = \exists \mathbf{y}. q_1 \wedge \dots \wedge q_m,$$

where

$$q_i ::= \varphi \mid \Psi$$

$$\varphi ::= C(t) \mid r(t_1, t_2)$$

$$\Psi ::= \varphi \mid \Psi_1 \wedge \Psi_2 \mid \circ^- \Psi \mid \circ \Psi \mid \Psi_1 \mathcal{S} \Psi_2 \mid \Psi_1 \mathcal{U} \Psi_2$$

and C is a concept description, r is a role name, t, t_1, t_2 are terms in $\text{Terms}(Q)$.

Derived temporal modalities like \diamond^- (*sometimes in the past*), \square^- (*always in the past*), \diamond , \square can be defined in a usual way (see, e.g. [4]).

Example 2. A query asking about students who had defended their thesis some time ago and had been ex-matriculated since then is expressed as follows:

$$Q(x) = \{x \mid \exists y. \diamond^- Student(x) \wedge exMatriculated(x) \mathcal{S} hasDefended(x, y)\}$$

The semantics of the TCQ is defined as follows.

Definition 4. A total function $\pi : \text{Terms}(\Psi) \rightarrow \Delta$ is a binding for a query Ψ in an interpretation \mathcal{I} , if $\pi(a) = a$ for all individuals $a \in \text{dom}(\pi)$, and the validity $\mathcal{I}, \pi \models \Phi$ for atemporal CQ φ is defined inductively:

$$\begin{aligned} \mathcal{I}, \pi \models C(t) & \quad \text{iff } \mathcal{I} \models C(\pi(t)) \\ \mathcal{I}, \pi \models r(t_1, t_2) & \quad \text{iff } \mathcal{I} \models r(\pi(t_1), \pi(t_2)) \\ \mathcal{I}, \pi \models \varphi_1 \wedge \varphi_2 & \quad \text{iff } \mathcal{I}, \pi \models \varphi_1 \text{ and } \mathcal{I}, \pi \models \varphi_2 \\ \mathcal{I}, \pi \models \exists y \varphi & \quad \text{iff } \exists e \in \Delta : \pi' = \pi[y/e] \text{ and } \mathcal{I}, \pi' \models \varphi \end{aligned}$$

where the notation $\pi[y/e]$ represents a binding π extended with $\pi(y) = e$ if y is not in the domain of π , otherwise the original value for y is replaced by e .

The validity for a TCQ Ψ and a KB $\mathcal{K} = \langle \mathcal{T}, (\mathcal{A}_i)_{0 \leq i \leq n} \rangle$ is extended as follows:

$$\begin{aligned} \mathcal{K}, i, \pi \models \varphi & \quad \text{iff } \forall \mathcal{I} \models_{\mathcal{T}} \mathcal{A}_i. \mathcal{I}, \pi \models \varphi \\ \mathcal{K}, i, \pi \models \Psi_1 \wedge \Psi_2 & \quad \text{iff } \mathcal{K}, i, \pi \models \Psi_1 \text{ and } \mathcal{K}, i, \pi \models \Psi_2 \\ \mathcal{K}, i, \pi \models \circ \Psi & \quad \text{iff } i < n \text{ and } \mathcal{K}, i+1, \pi \models \Psi \\ \mathcal{K}, i, \pi \models \circ^- \Psi & \quad \text{iff } i > 0 \text{ and } \mathcal{K}, i-1, \pi \models \Psi \\ \mathcal{K}, i, \pi \models \Psi_1 \mathcal{U} \Psi_2 & \quad \text{iff } \exists k, i \leq k \leq n : \mathcal{K}, k, \pi \models \Psi_2 \\ & \quad \text{and } \forall j, i \leq j < k : \mathcal{K}, j, \pi \models \Psi_1 \\ \mathcal{K}, i, \pi \models \Psi_1 \mathcal{S} \Psi_2 & \quad \text{iff } \exists k, 0 \leq k \leq i : \mathcal{K}, k, \pi \models \Psi_2 \\ & \quad \text{and } \forall j, k < j \leq i : \mathcal{K}, j, \pi \models \Psi_1 \end{aligned}$$

For a binding π , if, for every i , $\forall \mathcal{I} \models_{\mathcal{T}} \mathcal{A}_i \mathcal{I} \models \mathcal{K}$, this implies $\mathcal{I} \models \Psi$. If such evaluation exists, we write $\mathcal{K} \models \Psi$ and we say π is a match for Ψ in \mathcal{K} . For a tuple of individuals c_1, \dots, c_n mapped to a tuple of answer variables x_1, \dots, x_n we say c_1, \dots, c_n is a certain answer for Ψ in \mathcal{K} , iff $\mathcal{K} \models \Psi[x_1, \dots, x_n/c_1, \dots, c_n]$. We denote a set of certain answers for Ψ as $Ans(\Psi)$.

Definition 5. A union of temporal conjunctive queries (UTCQ) $Q(\mathbf{x})$ is a disjunction of temporal conjunctive queries (see Definition 3):

$$Q(\mathbf{x}) = \{\mathbf{x} \mid tconj_1(\mathbf{x}, \mathbf{c}) \vee \dots \vee tconj_n(\mathbf{x}, \mathbf{c})\}$$

4 Answering a Union of Temporal CQs Over a Sequence of ABoxes

4.1 Algorithm Answering a Union of Temporal CQs

The idea of answering a UTCQ against a set of ABoxes is to use temporal operators as the means of detection of time points at which atemporal CQs should be evaluated. Due to the recursive nature of such temporal operators as \mathcal{S} , \mathcal{U} we have to store all the ABoxes and the values of particular CQs depending on the operator. Intuitively, given $\Psi = \bigcirc \phi$ at a time point i , ϕ is evaluated at the time $i + 1$, and so on.

To be able to combine certain answers obtained from different CQs of one TCQ, let's take a closer look at the nature of certain answers.

A certain answer to a CQ ϕ is a binding π of each $x_i \in \mathbf{x}$ (distinguished variables) to some individual name that appeared in the KB \mathcal{K} , such that in all models of \mathcal{K} , $\mathcal{K} \models \phi(\pi(\mathbf{x}))$. There could be more than one certain answer for a CQ ϕ , so further we shall consider a set of certain answers for a query $\phi(\mathbf{x})$. A correspondent set of matches for ϕ actually produces some k -ary relation, where k is the arity of a CQ ϕ .

A certain answer to a UCQ Φ is a combination of answers of CQs in Φ , i.e. $c_1 \cup \dots \cup c_n$ where n is a number of CQs in Φ . For such a combination there are two possible situations: (i) disjuncts ϕ_{j_1}, ϕ_{j_2} in UCQ Φ use pairwise disjoint sets of distinguished variables (i.e. there are no common distinguished variables in two arbitrary disjuncts of Φ); (ii) some disjuncts can share (some) distinguished variables of each other. To deal with sets of certain answers (that are actually relations) we adopt two operators of relational algebra, namely, \times - a cross-product, and \bowtie - a natural join.

Cross-product operator \times [5] is used for the case (i).

Definition 6. Given two bindings $\pi_1 : (x_1, \dots, x_n) \rightarrow \Delta$, $\pi_2 : (y_1, \dots, y_m) \rightarrow \Delta$, their cross-product, $\pi_1 \times \pi_2$ is a binding $\pi : X \rightarrow \Delta$ where x, y are free variables that do not have any variables in common, and $X = (x_1, \dots, x_n, y_1, \dots, y_m)$.

Join operator \bowtie [5] is used for the case (ii) to join two bindings w.r.t. common variables in both bindings are mapped to same constant.

Definition 7. Given two bindings $\pi_1 : (x_1, \dots, x_n, z) \rightarrow \Delta$, $\pi_2 : (y_1, \dots, y_m, z) \rightarrow \Delta$, their join, $\pi_1 \bowtie \pi_2$ is a binding $\pi : X \rightarrow \Delta$ where x, y, z are free variables and $X = (x_1, \dots, x_n, y_1, \dots, y_m, z)$, iff every common variable z must be mapped to same constant $c \in \Delta$.

A correspondent binding for Φ will be: for (i) $\pi = \pi_{\phi_1} \times \dots \times \pi_{\phi_n}$, and for (ii) $\pi = \pi_{\phi_1} \bowtie \dots \bowtie \pi_{\phi_n}$

The following theorems show applications of \times and \bowtie for bindings.

Theorem 1. *Given a formula $\Phi = \phi_1 \wedge \phi_2$, where ϕ_1, ϕ_2 are CQ formulas, a binding $\pi = \pi_1 \bowtie \pi_2$ is a match for Φ iff bindings π_1, π_2 are matches for ϕ_1, ϕ_2 .*

Proof. It is true based on the definition of the join operator. \square

Theorem 2. *Given a formula $\Phi = \phi_1 \vee \phi_2$, where ϕ_1, ϕ_2 are CQ formulas, a binding $\pi = \pi_1 \times \pi_2$ is a match for Φ iff the binding π_1 is a match for ϕ_1 or the binding π_2 is a match for ϕ_2 .*

Proof. The \Rightarrow direction is trivial.

For \Leftarrow direction, assume $\pi_1 : (x_1, \dots, x_n, z) \mapsto \Delta, \pi_2 : (y_1, \dots, y_m, z) \mapsto \Delta$, and they are matches for ϕ_1 and ϕ_2 . From the nature of disjunction, we know that formula Φ is satisfiable if either ϕ_1 or ϕ_2 is satisfiable. That means if there is a match for either ϕ_1 or ϕ_2 . If z appears in both of the CQs, renaming z in one of the CQs does not change the validity. Therefore, we have that $\pi : (x_1, \dots, x_n, z, y_1, \dots, y_m, z') \mapsto \Delta$ which is obtained from $\pi_1 \times \pi_2$ is indeed a match for Φ . \square

Now, consider a structure of a certain answer to a union of temporal CQs (UTCQ). It is a combination of answers to a (set of) TCQ obtained at proper time points, referred by temporal operators used in a UTCQ.

One more thing to be explicitly addressed is that known algorithms for conjunctive query answering, such as [6], [7], are focused on query entailment, that is, a Boolean conjunctive query answering. This means that the task of answering an atemporal CQ requires a preprocessing step, and considers a Boolean conjunctive query answering algorithm as a black box. Namely, at the preprocessing step a *candidate match* (a tuple of variables, substituted via some binding π with a tuple of individuals c) is submitted to a Boolean conjunctive query answering engine, and that engine decides if such a candidate match is a certain answer.

Now, present the algorithm informally.

Eliminate temporal operators in a UTCQ The important step in our algorithm is to get a normal form where the temporal operators are used to decide at which time point should each CQ be evaluated. This is done by iterative application of the expansion rules Table 1. For every \circ and \circ^- operators, we just shift one point forward and backward. By doing these, we obtain a query that is in normal form whose atoms are UCQs, except some recursion atom which is a TCQ.

Replace the boolean operators with relational operators Every conjunction is replaced with join and every disjunction - with cross-product.

Retrieve an answer Use an arbitrary query answering algorithm [6–9] as a black-box approach to compute a set of answers for a given UCQ. If the original UTCQ contains $\mathcal{U}, \mathcal{S}, \square, \square^-, \diamond, \diamond^-$, the normal form of the transformed query might contain a recursion. In such case, if the time point $i < 0$ or $i > n$, then return \emptyset , else evaluate CQs with leading \circ or \circ^- for \mathcal{U}, \mathcal{S} and for derived modalities (if any).

Algorithm 1 Decide Q

Input: $\mathcal{K} = \{\mathcal{T}, (\mathcal{A}_i)_{0 \leq i \leq n}\}$: knowledge base consists of a TBox and a sequence of ABoxes at a time point $i, 0 \leq i \leq n$

Q : a UTCQ

Output: $Ans(Q, i)$ - a set of certain answers to Q at time point i

$Ans(Q, i) = \emptyset$

repeat

$Ans' = Ans(Q, i)$

if $Q = TCQ_1 \vee TCQ_2$ **then**

$Ans(Q, i) = Ans(TCQ_1, i) \times Ans(TCQ_2, i)$

end if

if $Q = TCQ_1 \wedge TCQ_2$ **then**

$Ans(Q, i) = Ans(TCQ_1, i) \bowtie Ans(TCQ_2, i)$

end if

if $Q = \circ^- TCQ$ **then**

if $i=1$ **then**

$Ans(Q, i) = \emptyset$

else

$Ans(Q, i) = Ans(TCQ, i - 1)$

end if

end if

if $Q = \circ TCQ$ **then**

if $i=n$ **then**

$Ans(Q, i) = \emptyset$

else

$Ans(Q, i) = Ans(TCQ, i + 1)$

end if

end if

if $Q = TCQ_1 \cup TCQ_2$ **then**

if $i=n$ **then**

$Ans(Q, i) = Ans(TCQ_2, i)$

else

$Ans(Q, i) = Ans(TCQ_2, i) \times (Ans(TCQ_1, i) \bowtie Ans(Q, i + 1))$

end if

end if

if $Q = TCQ_1 \mathcal{S} TCQ_2$ **then**

if $i=1$ **then**

$Ans(Q, i) = Ans(TCQ_2, i)$

else

$Ans(Q, i) = Ans(TCQ_2, i) \times (Ans(TCQ_1, i) \bowtie Ans(Q, i - 1))$

end if

end if

until $Ans' = Ans(Q, i)$

return $Ans(Q, i)$

Table 1. Equivalence rules of LTL for future operators. Taken from [4]

idempotent rule	$\Box\Psi \equiv \Box\Box\Psi$
	$\Diamond\Psi \equiv \Diamond\Diamond\Psi$
	$\Psi_1 \mathcal{U} (\Psi_1 \mathcal{U} \Psi_2) \equiv \Psi_1 \mathcal{U} \Psi_2$
	$(\Psi_1 \mathcal{U} \Psi_2) \mathcal{U} \Psi_2 \equiv \Psi_1 \mathcal{U} \Psi_2$
commutativity rule	$\Box \circ \Psi \equiv \circ \Box \Psi$
	$\Diamond \circ \Psi \equiv \circ \Diamond \Psi$
	$\circ(\Psi_1 \mathcal{U} \Psi_2) \equiv (\circ\Psi_1 \mathcal{U} \circ\Psi_2)$
distributivity rule	$\Box(\Psi_1 \wedge \Psi_2) \equiv (\Box\Psi_1 \wedge \Box\Psi_2)$
	$\Diamond(\Psi_1 \vee \Psi_2) \equiv (\Diamond\Psi_1 \vee \Diamond\Psi_2)$
	$\circ(\Psi_1 \wedge \Psi_2) \equiv (\circ\Psi_1 \wedge \circ\Psi_2)$
	$\circ(\Psi_1 \vee \Psi_2) \equiv (\circ\Psi_1 \vee \circ\Psi_2)$
	$((\Psi_1 \wedge \Psi_2) \mathcal{U} \Psi_3) \equiv ((\Psi_1 \mathcal{U} \Psi_3) \wedge (\Psi_2 \mathcal{U} \Psi_3))$
	$(\Psi_1 \mathcal{U} (\Psi_2 \vee \Psi_3)) \equiv ((\Psi_1 \mathcal{U} \Psi_2) \vee (\Psi_1 \mathcal{U} \Psi_3))$
temporal recursion rule	$\Box\Psi \equiv \Psi \wedge \Box\Psi$
	$\Diamond\Psi \equiv \Psi \vee \Diamond\Psi$
	$\Psi_1 \mathcal{U} \Psi_2 \equiv \Psi_2 \vee (\Psi_1 \wedge \circ(\Psi_1 \mathcal{U} \Psi_2))$
absorption rule	$\Diamond\Box\Psi \equiv \Box\Diamond\Psi$
	$\Box\Diamond\Psi \equiv \Diamond\Box\Psi$

In Table 1, presented are some equivalence rules in LTL, used in Algorithm 1.

For the illustration of Algorithm 1 consider some examples, assuming that Algorithm 1 returns a set Ans of answers to Ψ at the time point i .

Example 3. Given a TCQ query $\Psi = \circ^-(\Phi_1 \mathcal{U} \Phi_2)$ at a point i .

$$\begin{aligned}
Ans(\Psi, i) &= Ans(\circ^-(\Phi_1 \mathcal{U} \Phi_2), i) \\
&\quad * \backslash \text{move back one point by } \circ^- \\
&= Ans(\Phi_1 \mathcal{U} \Phi_2, i-1) \\
&\quad * \backslash \text{expansion rule for } \mathcal{U} \\
&= Ans(\Phi_2 \vee (\Phi_1 \wedge \circ\Psi), i-1) \\
&\quad * \backslash \text{transforming } \vee \\
&= Ans(\Phi_2, i-1) \times Ans(\Phi_1 \wedge \circ\Psi, i-1) \\
&\quad * \backslash \text{transforming } \wedge \\
&= Ans(\Phi_2, i-1) \times (Ans(\Phi_1, i-1) \bowtie Ans(\circ\Psi, i-1)) \\
&\quad * \backslash \text{move forward one point by } \circ \\
&= Ans(\Phi_2, i-1) \times (Ans(\Phi_1, i-1) \bowtie Ans(\Psi, i))
\end{aligned}$$

If $i = 0$ in $Ans(\Phi_2, i-1)$ and $Ans(\Phi_1, i-1)$, then the evaluation of Ψ is the empty set.

A more complex example is given below.

Example 4. Given a TCQ query $\Psi = \diamond^-(\Phi_1 \mathcal{U} \Phi_2)$ at a point i .

$$\begin{aligned}
 \text{Ans}(\Psi, i) &= \text{Ans}(\diamond^-(\Phi_1 \mathcal{U} \Phi_2), i) \\
 &\quad * \backslash \text{expansion rule for } \diamond^- \\
 &= \text{Ans}((\Phi_1 \mathcal{U} \Phi_2) \vee \circ^-\Psi, i) \\
 &\quad * \backslash \text{transforming } \vee \\
 &= \text{Ans}(\Phi_1 \mathcal{U} \Phi_2, i) \times \text{Ans}(\circ^-\Psi, i) \\
 &\quad * \backslash \text{move back one point by } \circ^- \\
 &= \text{Ans}(\Phi_1 \mathcal{U} \Phi_2, i) \times \text{Ans}(\Psi, i - 1) \\
 &\quad * \backslash \text{We substitute } \Phi_1 \mathcal{U} \Phi_2 \text{ with } \Psi' \\
 &\quad \text{Expansion rule for } \mathcal{U} \\
 &= \text{Ans}(\Phi_2 \vee (\Phi_1 \wedge \circ\Psi'), i) \times \text{Ans}(\Psi, i - 1) \\
 &\quad * \backslash \text{transforming } \vee \\
 &= \text{Ans}(\Phi_2, i) \times \text{Ans}(\Phi_1 \wedge \circ\Psi', i) \times \text{Ans}(\Psi, i - 1) \\
 &\quad * \backslash \text{transforming } \wedge \\
 &= \text{Ans}(\Phi_2, i) \times \\
 &\quad \left(\text{Ans}(\Phi_1, i) \bowtie \text{Ans}(\circ\Psi', i) \right) \times \text{Ans}(\Psi, i - 1) \\
 &\quad * \backslash \text{move forward one point by } \circ \\
 &= \text{Ans}(\Phi_2, i) \times \left(\text{Ans}(\Phi_1, i) \bowtie \text{Ans}(\Psi', i + 1) \right) \times \text{Ans}(\Psi, i - 1)
 \end{aligned}$$

If $i = n$ in $\text{Ans}(\Psi', i + 1)$, then $\text{Ans}(\Psi', i + 1)$ is evaluated to the empty set.

There is one thing we have to ensure that in the intersection of two sets of answers for conjunction of CQs a certain answer is obtained, i.e. there is a common answer for both CQs, otherwise an empty set. One way to do this is to retrieve all answers for each CQ and then to intersect them to get some common answers. Another way is first to retrieve an answer of a UCQ and then to decide if this answer is also the answer for the other CQs in the conjunction, otherwise keep retrieving and deciding until there is no more answer obtained. The former way is preferred since it offers more practical solution. It means that we can deal with it using relational algebra operators or database language operators.

4.2 Termination, Soundness, Completeness of the Algorithm

Definition 8. (*UTCQ closure*). Given a temporal union of conjunctive queries Q , its closure set, $Cl(Q)$ is a set of query atoms closed under the following rules

$$\begin{aligned}
 \text{if } q \in Q &\quad \text{then } q \in Cl(Q) \\
 \text{if } \circ^-q \in Q &\quad \text{then } q \in Cl(Q) \\
 \text{if } \circ q \in Q &\quad \text{then } q \in Cl(Q) \\
 \text{if } q_1 \wedge q_2 &\quad \text{then } q_1, q_2 \in Cl(Q) \\
 \text{if } q_1 \vee q_2 &\quad \text{then } q_1, q_2 \in Cl(Q) \\
 \text{if } q_1 \mathcal{U} q_2 &\quad \text{then } q_1, q_2, \circ(q_1 \mathcal{U} q_2) \in Cl(Q) \\
 \text{if } q_1 \mathcal{S} q_2 &\quad \text{then } q_1, q_2, \circ^-(q_1 \mathcal{S} q_2) \in Cl(Q)
 \end{aligned}$$

Since a closure set for a UTCQ is finite, Algorithm 1 terminates after a finite number of steps.

Theorem 3. (Local) termination. Given a UTCQ Q and a knowledge base $\mathcal{K} = \{\mathcal{T}, (\mathcal{A}_i)_{0 \leq i \leq n}\}$. Algorithm 1 always terminates.

Proof. We can show the local termination inductively.

Base case. Any query is also contained in the closure set of itself.

Inductive case.

$(C(a), r(a_1, a_2))$ If we have a query Q which is atomic, then the closure set contains $C(a)$ or $r(a_1, a_2)$.

(\circ^-TCQ) For such query $Cl(Q) = \{TCQ, \circ^-TCQ\}$, i.e. evaluated are two elements, and in case of $i = 0$ the value of \circ^-TCQ is known to be \emptyset , so Algorithm 1 stops after two evaluations.

$(TCQ_1 \cup TCQ_2)$ For such query $Cl(Q) = \{TCQ_2, TCQ_1, TCQ_1 \cup TCQ_2, \circ(TCQ_1 \cup TCQ_2)\}$

$(TCQ_1 \mathcal{S} TCQ_2)$ For such query $Cl(Q) = \{TCQ_2, TCQ_1, TCQ_1 \mathcal{S} TCQ_2, \circ(TCQ_1 \mathcal{S} TCQ_2)\}$ \square

Theorem 4. Soundness. If for UTCQ Q its answer set $Ans(Q(x), i)$, obtained with Algorithm 1, is not empty, then Q has at least those certain answers that are in $Ans(Q, i)$.

Proof. We prove by induction. We start with evaluating non-temporal query, i.e. a query containing no temporal operator.

Base case If we have an atomic query in the form of $C(a)$, then using any approach of CQ answering we obtain all the answers for the query Q entailment over $\mathcal{K} = \{\mathcal{T}, (\mathcal{A}_i)_{0 \leq i \leq n}\}$. If $\mathcal{K} \models C(a)$ and $a \in Ans(Q(x), i)$, the function returns a and this value is stored in $Ans(Q(x), i)$. By Definition 4, this result tells us that the individual a is a certain answer to the query $C(x)$ w.r.t. the match $\pi(x) = a$. The same result is obtained if we have atomic query in the form of $r(a, b)$.

Inductive case can be obtained by Definition 4. \square

Theorem 5. Completeness. If a UTCQ Q has a certain answer ans , then Algorithm 1 shows that this answer is in $Ans(Q, i)$.

Proof. By contradiction. Assume that (i) $Q(x)$ has a certain answer ans w.r.t π , (ii) $Ans(Q, i)$ - is a set of certain answers obtained by Algorithm 1, and (iii) $ans \notin Ans(Q, i)$. By (i), we know that $\mathcal{K} \models Q(ans)$ and that for all time points $0 \leq i \leq n$ in all models \mathcal{I} , such that $\mathcal{I} \models \mathcal{K}$, $\mathcal{I} \models Q(ans)$. By (ii), for Algorithm 1 to return $Ans(Q, i)$ such that $ans \notin Ans(Q, i)$ there are several reasons for it.

Q is atomic. If Q is atomic, i.e. in the form $C(x)$ or $r(x, y)$, then we know that $Ans(Q, i)$ does not contain ans . This means that there is a model \mathcal{I} of a knowledge base \mathcal{K} which does not entail $Q(ans)$. But this is a contradiction to our assumption (i).

$(TCQ_1 \wedge TCQ_2)$. If $Ans(Q, i)$ does not contain ans , according to Algorithm 1 it means that $ans \notin Ans(TCQ_1, i) \bowtie Ans(TCQ_2, i)$. This, in turn, leads to the existence of a model \mathcal{I} of a knowledge base \mathcal{K} such that $\mathcal{I} \models TCQ_1(ans)$ and $\mathcal{I} \not\models TCQ_2(ans)$ or vice versa, that contradicts to (i).

$(TCQ_1 \vee TCQ_2)$. If $Ans(Q, i)$ does not contain ans , according to Algorithm 1 it means that $ans \notin Ans(TCQ_1, i) \times Ans(TCQ_2, i)$. This, in turn, leads to the existence of a model \mathcal{I} of a knowledge base \mathcal{K} such that either $\mathcal{I} \not\models TCQ_1(ans)$ or $\mathcal{I} \not\models TCQ_2(ans)$, that contradicts to (i).

(\circ^-TCQ). If $Ans(Q, i)$ does not contain ans , according to Algorithm 1 it means that $ans \notin Ans(Q, i - 1)$. This, in turn, leads to the existence of a model \mathcal{I} of a knowledge base \mathcal{K} such that $\mathcal{I}, i - 1 \not\models Q(ans)$, that contradicts to (i).

($TCQ_1 \cup TCQ_2$). If $Ans(Q, i)$ does not contain ans , according to Algorithm 1 it means that $ans \notin Ans(TCQ_2, i) \times (Ans(TCQ_1, i) \bowtie Ans(Q, i + 1))$. This, in turn, leads to the existence of a model \mathcal{I} of a knowledge base \mathcal{K} such that either $\mathcal{I}, i \not\models TCQ_2(ans)$ or $\mathcal{I}, i \not\models TCQ_1(ans)$ and $\mathcal{I}, i + 1 \not\models Q$, that contradicts to (i).

($TCQ_1 \mathcal{S} TCQ_2$). If $Ans(Q, i)$ does not contain ans , according to Algorithm 1 it means that $ans \notin Ans(TCQ_2, i) \times (Ans(TCQ_1, i) \bowtie Ans(Q, i - 1))$. This, in turn, leads to the existence of a model \mathcal{I} of a knowledge base \mathcal{K} such that either $\mathcal{I}, i \not\models TCQ_2(ans)$ or $\mathcal{I}, i \not\models TCQ_1(ans)$ and $\mathcal{I}, i - 1 \not\models Q$, that contradicts to (i).

The proof for the temporal operator \circ acting in the direction of future can be completed in the same manner. \square

5 Related Work and Conclusions

Transition graphs for a temporal query language answering over a finite set of versions of a database were investigated in [10]. The expressivity of a temporal query language presented is however restricted either to past [11], or to future [10], [12] direction of time. Known are several algorithms for answering unions of conjunctive queries over knowledge bases with static TBox and ABox, for example works of Ortiz [6], Glimm [7], Tessaris [9], Motik [8] should be mentioned. Any of those algorithms could serve as a basis for finding answers to atemporal CQs at particular time points, whereas possible extensions of those algorithms for the application to a sequence of ABoxes is an open question. A language of temporal conjunctive queries with negation, together with the computational and combined computational complexity is introduced in [13]. Summing up, obtaining benefits from keeping a large evolving ABox of a knowledge base in a database and applying TBox of that knowledge base to obtain missing assertional axioms is one of the ways of dealing with complex evolving domains. It is interesting, due to high computational complexity of temporal conjunctive query answering in general, to find a balance between the expressivity of a query language and its practical applicability.

Acknowledgements The presented results were obtained during the research visit of the author to the Chair of Automata Theory at Dresden University of Technology. The author is grateful to the group of Prof. Franz Baader, and in particular, Eldora, Marcel Lippmann and Anni-Yasmin Turhan for the fruitful discussions and ideas at the stage of early drafts of the paper.

References

1. Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Rosati R. Linking Data to Ontologies. J. on Data Semantics, X, 133–173 (2008)

2. Baader, F., Bauer, A., Baumgartner, P., Cregan, A., Gabaldon, A., Ji, K., Lee, K., Rajaratnam, D., Schwitter, R. A novel architecture for situation awareness systems. In: Giese, M. and Waaler, A. (eds.) Proc. 18th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (Tableaux 2009). LNCS, vol. 5607, pp. 77–92. Springer, Berlin/Heidelberg (2009)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.). The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
4. Baier, C., Katoen, J.-P. Principles of Model Checking. The MIT Press, Cambridge, Massachusetts, USA (2008)
5. Abiteboul, S., Hull, R., Vianu, V. Foundations of Databases. Addison-Wesley (1995)
6. Ortiz de la Fuente, M.M. Query Answering in Expressive Description Logics Techniques and Complexity Results. PhD Thesis. Technische Universität Wien, Fakultät für Informatik (2010)
7. Glimm, B. Querying Description Logic Knowledge Bases. PhD Thesis. The University of Manchester (2007)
8. Motik, B. Reasoning in Description Logics using Resolution and Deductive Databases. Universität Karlsruhe (2006)
9. Tessaris, S. Questions and answers: reasoning and querying in Description Logic. The University of Manchester (2001)
10. Lipeck, U.W. Transformation of Dynamic Integrity Constraints into Transaction Specifications. Theor. Comput. Sci., 76(1), pp. 115–143 (1990)
11. Schwiderski, S., Hartmann, T., Saake, G. Monitoring Temporal Preconditions in a Behaviour Oriented Object Model. Data Knowl. Eng., 14(2), pp. 143–186 (1994)
12. Lipeck, U.W., Feng, D. Construction of Deterministic Transition Graphs from Dynamic Integrity Constraints. LNCS, vol. 344, pp. 166–179. Springer Verlag (1989)
13. Baader F., Borgwardt, S., Lippmann, M. On the Complexity of Temporal Query Answering. Technical report LTCS-Report 13-01. Available at <http://lat.inf.tu-dresden.de/research/reports/2013/BaBoLi-LTCS-13-01.pdf> (2013)

An Adaptive Forecasting of Nonlinear Nonstationary Time Series under Short Learning Samples

Elena Mantula¹ and Vladimir Mashtalir¹

¹ Kharkiv National University of Radio Electronics, informatics department
Lenin ave., 14, 61166, Kharkiv, Ukraine

Mashtalir@kture.kharkov.ua, ElenaMantula@gmail.com

Abstract. Methods of nonstationary nonlinear time series forecasting under bounded a priori information provide an interdisciplinary applications area that is concerned with learning and adaptation of solutions from a traditional artificial intelligence point of view. It is extremely difficult to solve this type of problems in its general form, therefore, an approach based on the additive nonlinear auto regressive model with exogenous inputs and implemented on the base of parallel adalines set has been proposed. To find optimal combination of forecasts, an improvement of global random search has been suggested.

Keywords. Neural networks, forecasting model, combination of forecasts

Key terms. Environment, MathematicalModel

1 Introduction

‘Conscious’ decision making, in all possible varieties, is perhaps the most principal goal of artificial intelligence systems. Necessary ‘creativity’ implies the ability to produce novel solutions which are better than previous ones. The computational tools that assist in decision making should be such that they should take into all aspects of dissimilarity between a priori and a posteriori uncertainty. Uncertainty account is, per se, a manifestation of information deficiency, and relevant information is, on the contrary, a capacity to reduce uncertainty. An elimination of such rich in content gaps provides groundwork of knowledge engineering and management. In machine intelligence, manifold forecasts can be used for knowledge producing. The goal of the paper consists in reasonable (perfectly optimal) combination of forecasts to provide reliable semantic interpretation of achieved results with purpose knowledge generation.

Nowadays mathematical forecasting models of the behavior of objects, systems and phenomena in a wide variety of applications are well understood. There is a wealth of publications on this subject. It should be noted that the behavior of the objects is often given in the form of time series. Thus to forecast its behavior a variety of approaches to the analysis of time series can be used. Such approaches can be either traditional statistical methods (regression, correlation, spectral, Box-Jenkins) or adaptive, based on an exponential smoothing, tuning or learning forecasting models, or

intellectual, using various neural networks.

At present there are many objects (financial, economical, biomedical, etc.), described by time series containing unknown behavior trends, seasonal components, stochastic and random components, which significantly complicate synthesis of an effective predictive model. This complexity is especially pronounced in the environmental monitoring problems [1], where the analyzing time series have in equal measure stochastic and chaotic type of changes, have apparent nonstationarity and are subjected to striking changes.

In these conditions artificial ccc have proved to be useful tools in the best way [2-13]. As a rule, they realize so-called NARX-model [14], which has the form

$$\hat{y}(k) = f(y(k-1), \dots, y(k-n_A), x(k-1), \dots, x(k-n_B)) \quad (1)$$

where $\hat{y}(k)$ is an estimation of forecasted variable $y(k)$ at discrete time $k=1,2,\dots$; $f(\circ)$ denotes certain nonlinear transform which is realized by a neural network; $x(k)$ is the observed exogenous factor that influences the behavior of $y(k)$; n_A, n_B are observations memory parameters.

Moreover, it is not a matter of available observations insufficiency, since properties of time series (e.g. such indicator as air pollution in ecological forecasting) are changed so often that a neural network does not have time to detect separate stationary parts. In this connection there is a need to construct based on the neural network approach simplified predictive models for training which require the small enough volume data set.

2 Synthesis of a forecasting model

In conditions of input data lack instead of NARX-model (1) it is appropriate to use the so-called ANARX-model introduced in [15, 16] and fully investigated in [17, 18]. In general ANARX-model can be written as

$$\begin{aligned} \hat{y}(k) &= f_1(y(k-1), x(k-1)) + f_2(y(k-2), x(k-2)) + \dots \\ &+ f_{\max\{n_A, n_B\}}(y(k-n_A), x(k-n_B)) = \\ &= \sum_{l=1}^{\max\{n_A, n_B\}} f_l(y(k-l), x(k-l)) \end{aligned} \quad (2)$$

where original task is decomposed into many local ones with two input variables $y(k-l), x(k-l), l=1,2,\dots,\max\{n_A, n_B\}$.

For such nonlinear transforms it is quite convenient to use so-called N-adaline (abbr.: adaptive linear element) [19-21] that provide quadratic approximation of the data sequence. Fig. 1(a) demonstrates the architecture of N-adaline and (b) illustrates the architecture of ANARX-model constructed using N-adaline.

As we can see, N-adaline represents a generally accepted two-input adaline with a nonlinear preprocessor formed by three blocks of the product (Π) and the evaluator of the quadratic combination in the form

$$f_l(y(k-l), x(k-l)) = w_{l0} + w_{l1}y(k-l) + w_{l2}y^2(k-l) + w_{l3}y(k-l)x(k-l) + w_{l4}x^2(k-l) + w_{l5}x(k-l)$$

where each N-adaline contains 6 synaptic weights w_{lp} , $l=1,2,\dots,\max\{n_A, n_B\}$, $p=0,1,\dots,5$. As a matter of fact, ANARX-model is formed by two lines of delay elements z^{-1} and $\max\{n_A, n_B\}$ parallel learned N-adaline.

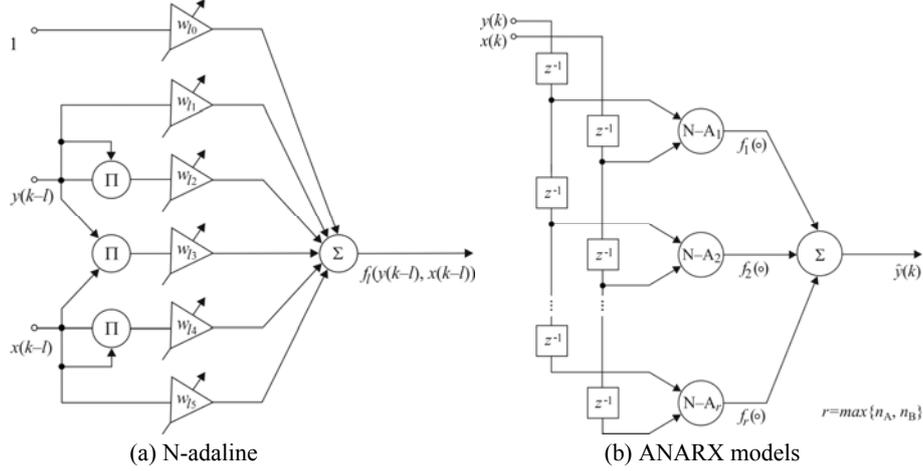


Fig. 1. N-adaline and ANARX models based on N-adalines

Each from N-adalines is configured with any of the linear learning algorithms [22], however, it is clear that a limited amount of a priori information requires the use of time-optimal procedures. As such can be, for example, adaptive-multiplicative modification of Kachmarz adaptive algorithm [23], which assumes in this case the form

$$w_l(k) = w_l(k-1) + \gamma \frac{y(k) - w_l^T(k-1)\phi_l(k)}{\beta + \|\phi_l(k)\|^2} \phi_l(k) \tag{3}$$

where $w_l = (w_{l0}, w_{l1}, w_{l2}, w_{l3}, w_{l4}, w_{l5})^T$; $\phi_l(k) = (1, y(k-l), y^2(k-l), y(k-l)x(k-l), x^2(k-l), x(k-l))^T$; $0 < \gamma < 2$, $\beta \geq 0$ are some algorithm parameters selected on the base of empirical reasons.

If the data sequences are ‘contaminated’ by perturbations, instead of the one-step algorithm (3) it is profitably to apply procedures that provide filtering of perturbation and at the same time they have to be suitable for using in non-stationary conditions. It should be noted that modification of the recursive least squares method on a sliding window can be used [24]. The traditional estimation method of least squares on the window with s observations has the form

$$w_l(k) = (\sum_{\tau=k-s+1}^k \phi_l(\tau)\phi_l^T(\tau))^{-1} \sum_{\tau=k-s+1}^k \phi_l(\tau)y(\tau)$$

and recurrent one can be presented as

$$\begin{cases} P(k) = P_s(k-1) - \frac{P_s(k-1)\varphi_I(k)\varphi_I^T(k)P_s(k-1)}{1 + \varphi_I^T(k)P_s(k-1)\varphi_I(k)}, \\ P_s(k) = P(k) + \frac{P(k)\varphi_I(k-s)\varphi_I^T(k-s)P(k)}{1 - \varphi_I(k-s)P(k)\varphi_I(k-s)}, \\ p_s(k) = p_s(k-1) + \varphi_I(k)y(k) - \varphi_I(k-s)y(k-s), \\ w_I(k) = P_s(k)p_s(k). \end{cases} \quad (4)$$

We also note that if the algorithm (3) is in fact time-optimal gradient procedure, then the algorithm (4) is produced by Gaussian-Newton optimization procedure.

3 Optimal combination of forecasts

In real conditions the choice of the forecasting model structure is not a trivial task, especially that the same time series can be effectively described by a variety of different models. Also, the value of the lag orders n_A, n_B remains unknown what makes it necessary to consider a set of competing models, and nonstationarity of analyzed series necessitates the use of various learning algorithms (in this case, (3), (4)) with different values γ, β, s . Thus, there arises a set of forecasts of the same process, from which we have to select the best.

To find the best forecast it is possible to use sufficiently effective approach, based on the optimal combination of forecasts [25], under which optimal in the sense of given criterion J^c linear combination is searching for a set of existing forecasts of the same series $\hat{y}_j(k)$, $j = 1, 2, \dots, m$

$$\hat{y}(k) = \sum_{j=1}^m c_j \hat{y}_j(k) \quad (5)$$

where the parameters of the combination satisfy the condition of unbiasedness

$$\sum_{j=1}^m c_j = 1. \quad (6)$$

In [25], an analytical approach to the weights c_j finding in (5) by optimizing the sum of squared errors criterion for forecasting with the constraints (6) is proposed. The use of one-step squared forecast errors criterion leads to the estimation

$$c_j(k) = \frac{\hat{y}_j(k)}{\sum_{j=1}^m \hat{y}_j(k)}.$$

However, combining of the analytical parameter estimates can be obtained under application of standard quadratic criterion J^c solely that specified by linearity of its derivatives so the solution of the problem reduces to solving a system of linear equations. At the same time for practitioners as a rule assess of the quality of forecasting

using the residual variance is unconvincing, and therefore characteristics allowing to estimate the accuracy in percentage are generally used, such as the criterion of a minimum of absolute percentage error

$$MAPE = \sum_{k=1}^N \left| \frac{y(k) - \hat{y}(k)}{y(k)} \right| 100\% \quad (7)$$

or maximum of the determination coefficient

$$R^2 = \left(1 - \frac{\sum_{k=1}^N (y(k) - \hat{y}(k))^2}{\sum_{k=1}^N (y(k) - \frac{1}{N} \sum_{k=1}^N y(k))^2} \right) 100\% . \quad (8)$$

It is obvious that in this case analytical estimations can not be obtained, and the use of gradient optimization procedures becomes more complicated due to sufficiently complex properties of functions (7), (8). In this connection the use of genetic algorithms is proposed in [26, 27]. Though such algorithms can find the global extremum, their own distinctive features are numerical awkwardness, they have a set of free parameters necessary defined by the user and at last it should be mentioned a low rate of convergence. Therefore, notice should be taken to more an efficient approach based on the random search [28] and its adaptive modifications. The most simple procedure, which allows to search for a global extremum, is walking random global search [28]. In general, this procedure is a statistic extension of the regular gradient search, and to provide the global search, random disturbance $\zeta(k)$ superimposes on character on a gradient movement what creates stochastic walking mode.

In the continuous case, the gradient method of minimization (maximization) of the goal function $J^c(t)$ is reduced to the motion of a point $c(t) = c_1(t), \dots, c_j(t), \dots, c_m(t)$ in m -dimensional space of adjustable parameters by a force directed toward the anti-gradient.

The trajectory of movement by antigradient $c(t)$ leads tuning process to a singular point. If starting point $c(0)$ belongs to an attraction region of global extremum then the corresponding trajectory will lead to a global minimum of the function $J^c(t)$. But if the point $c(0)$ does not belong to this region, the movement in the direction of anti-gradient will result in a local minimum, from which it is impossible to get out under the influence of forces directed by antigradient. Exactly because, it is helpful to use a random mechanism. Random shocks may help point $c(t)$ to overcome the barrier that separates the local minimum in which the learning process hit from the area in which the objective function $J^c(t)$ could further decrease. Under the influence of ‘skew’ toward anti-gradient and random shocks such movement is determined by the differential equation

$$\frac{dc(t)}{dt} = -\eta \nabla_c J^c(t) + \zeta(t)$$

where $\zeta(t)$ is m -dimensional normal random process with zero mathematical expect-

tation, delta-figurative autocorrelation function and components variance σ_ζ^2 ; η is parameter of step, ∇_c denotes gradient vector. It should be emphasized that for function (7) the components of the gradient can acquire the value $+1$ or -1 . Generally, this algorithm provides searching for a global extremum [29].

Searching for global extremum can be speed up by reasonable selection of σ_ζ^2 and an adaptation during this process can be introduced in two ways. First, under introducing inertia in the learning process, it is possible to get a search similar to the movement by the method of ‘heavy ball’ [30]. Such movement is described by the differential equation

$$\frac{d^2c(t)}{dt^2} + b \frac{dc(t)}{dt} = -\eta \nabla J^c(t) + \zeta(t) \quad (9)$$

where b is shockproofing coefficient (the more b , the less manifest of inserted inertia).

On time series processing, i.e. in discrete time, procedure (9) corresponds to the learning algorithm, described by the second order difference equation [31]

$$c(k) = c(k-1) + bc(k-2) - \eta(k) \nabla_c J^c(k) + \zeta(k) \quad (10)$$

coinciding under $b=0$ with walking random search. It is interesting to note that (10) is none other than the ARX- model of the second-order.

Second, the adaptation in the process of global search can be introduced by random process $\zeta(t)$ control, for example,

$$\frac{d\zeta(t)}{dt} = -\delta \zeta(t) - \eta_\zeta \frac{dJ^c(t)}{dt} + \sigma_\zeta^2 H(t) \quad (11)$$

where $\delta > 0$ is a autocorrelation parameter of random process $\zeta(t)$; $H(t)$ is a vector of flat random noise. Introduce a modification of (11) in the discrete form

$$\zeta(k) = (1 - \delta)\zeta(k-1) - \eta_\zeta(k) \Delta J^c(k) + \sigma_\zeta^2 H(k) \quad (12)$$

where Δ is the symbol of the first difference (discrete analogue of the derivative).

As it is easily seen from (11), (12), the optimization of the search process can be performed by appropriate selection of parameters δ , η_ζ and σ_ζ^2 , since each of them acts on the certain properties of the search. Indeed, variation of the autocorrelation parameter δ determines the rate of the process $\zeta(k)$ decay that regulates its relations with the past. Thus, one can have an influence upon a search making it more or less dependent on the previous history if it is necessary.

Some few words of comment are desirable for parameters δ and η_ζ interaction explanation. If the search step η_ζ determines the intensity of accumulation of learning experience, then δ characterizes the level of this experience forgetting during the search. In this sense, these parameters are antagonistic. If in general $\delta=0$ and there is no forgetting the vector $\eta_\zeta(k)$ increases in the direction of anti-gradient. Variance

of the process $\eta_{\zeta}(k)$ is determined by the value σ_{ζ}^2 and intensity of the flat random noise disturbance $H(k)$. If σ_{ζ}^2 is sufficiently large then search may become unstable and, at low value, global properties are worsening. Thus, the use of a modified global random search allows simplify significantly the process of linear combination $c_j(k)$, $j = 1, 2, \dots, m$ tuning.

4 Conclusion

The problem of nonstationary nonlinear time series forecasting under bounded a priori information has been considered. An approach based on the additive nonlinear autoregressive model with exogenous inputs and implemented on the base of parallel adalines set has been proposed. To find optimal combination of forecasts, an improvement of global random search has been suggested. Distinctive feature of the approach is the computational simplicity and high performance attained by significant reducing the number of adjustable parameters.

References

1. Zanetti, P.: Air Pollution Modelling. Van Nostrand Reinhold, New York (1990)
2. Reich, S.L., Gomez, D.R., Dawidowski, L.E.: Artificial Neural Network for the Identification of Unknown Air Pollution Sources. *Atmosphere Environment*, Vol. 33, pp. 3045-3052 (1999)
3. Perez, P., Trier, A., Reyes, J.: Prediction of PM_{2.5} Concentration Several Hours in Advance Using Neural Networks in Santiago, Chile. *Atmospheric Environmental*, Vol. 34, pp. 1189–1196 (2000)
4. Niska, N., Hiltunen, T., Karppinen, A., Ruuskanen, J., Kolehmanen, M.: Evolving the Neural Network Model for Forecasting Air Pollution Time Series. *Engineering Application of Artificial Intelligence*, Vol. 17, 159–167 (2004)
5. Corani G.: Air Quality Prediction in Milan: Feed-Forward Neural Networks, Pruned Neural Networks and Lazy Learning. *Ecological Modeling*, Vol. 185, pp. 513–529 (2005)
6. Athanasiadis, I.N., Karatzas, K.D., Mitkas, P.A.: Classification Techniques for Air Quality Forecasting. In: Brewka G., Coradeschi S., Perini A. and Traverso P. (eds.): *Proc. 17th European Conference on Artificial Intelligence*, IOS Press, Amsterdam, 4.1–4.7 (2006)
7. Perez, P., Reyes, J.: An Integrated Neural Network Model for PM₁₀ Forecasting. *Atmospheric Environment*. Vol. 40, pp. 2845–2857 (2006)
8. Lira, T.S., Barrozo, M.A.S., Assis, A.J.: Air Quality Prediction in Uberlandia, Brasil, Using Linear Models and Neural Networks. In: Plesu V., Agachi P. (eds.): *Proc. 17th European Symp. on Computer Aided Process Engineering*, Elsevier, Amsterdam, pp. 1–6 (2007)
9. Kurt, A., Gulbagci, B., Karaca, F., Alagha, O.: An Online Air Pollution Forecasting System Using Neural Networks. *Environmental International*, Vol. 34 (2008) 592–598
10. Carnevale, C., Finzi, G., Pisoni, E., Volta, M.: Neuro-Fuzzy and Neural Network Systems for Air Quality Control. *Atmospheric Environmental*, Vol. 43, pp. 4811–4821 (2009)

11. Nagendra, S.M., Shiva, Khare M.: Modelling Urban Air Quality Using Artificial Neural Network. *Clean Technical Environmental Policy*, Vol. 7, pp. 116–126 (2005)
12. Aktan, M, Bayraktar, H.: The Neural Network Modeling of Suspended Particulate Matter with Autoregressive Structure. *Ekoloji*, Vol. 19, No. 74, pp. 32–37 (2010)
13. Esau, I.: On Application of Artificial Neural Network Methods in Large-Eddy Simulations with Unresolved Urban Surfaces. *Modern Applied Science*, Vol. 4, No. 8, 3–11 (2010)
14. Nelles, O.: *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Springer, Berlin (2001)
15. Chowdhury, F.N., Input-Output Modeling of Nonlinear Systems with Time-Varying Linear Models. *IEEE Trans. on Automatic Control*, Vol. 45, No. 7, pp. 1355–1358 (2000)
16. Kotta, Ü., Sadegh, N.: Two Approaches for State Space Realization of NARMA Models: Bridging the Gap. *Mathematical and Computer Modeling of Dynamical Systems*, Vol. 8, No. 1, pp. 21–32 (2002)
17. Belikov, J., Vassiljeva, K., Petlenkov, E., Nomm S.: A Novel Taylor Series Based Approach for Control Computation in NN–ANARX Structure Based Control of Nonlinear Systems. In: *Proc. 27th Chinese Control Conference*, Beihang University Press, Kunming, pp. 474–478 (2008)
18. Vassiljeva, K., Petlenkov, E., Belikov, J.: State-Space Control of Nonlinear Systems Identified by ANARX and Neural Network Based SANARX Models. In: *Proc. WCCI 2010 IEEE World Congress on Computational Intelligence*, IEEE CSS, Piscataway, pp. 3816–3823 (2010)
19. Pham, D.T. Liu, X.: Modeling and Prediction Using GMDH Networks of Adalines with Nonlinear Preprocessors. *Int. J. System Science* Vol. 25, No. 11 (1994) 1743–1759
20. Pham, D.T. Liu, X.: *Neural Networks for Identification, Prediction and Control*. Springer, London (1995)
21. Rudenko, O.G., Bodyanskiy, Ie.V.: *Artificial Neural Networks*. SMIT, Kharkov (2005) (in Russian)
22. Haykyn, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall. Inc., New York (1999)
23. Raybman, N.S., Chadeev, V.M.: *Creating of Manufacture Process Models*. Energiya, Moscow (1975) (in Russian)
24. Perelman, I.I.: *Operative Identification of Control Objects*. Energoatomizdat, Moscow (1982) (in Russian)
25. Sharkeya, A.J.C.: On Combining Artificial Neural Nets. *Connection Science*, Vol. 8, No. 3, pp. 299–314 (1996)
26. Zagoryjko, N.G.: *Empirical Prediction*. Nauka, Novosibirsk (1979) (in Russian)
27. Zagoryjko, N.G.: *Applied Approach of Data Analysis*, p. 264 (1999) (in Russian)
28. Rastrigin, L.A.: *Statistical Search Technology*. Nauka, Moscow (1968) (in Russian)
29. Rastrigin, L.A.: *Systems of Extremal Control*. Nauka, Moscow (1974) (in Russian)
30. Polyak B.T.: *Introduction into Optimization*. Nauka, Moscow (1983) (in Russian)
31. Bodyanskiy, Ie. V., Rudenko, O.G.: *Artificial Neural Networks: Arhitectures, Learning, Applications*. TELETEx, Kharkov (2004) (in Russian)

Application of an Instance Migration Solution to Industrial Ontologies

Maxim Davidovsky¹, Vadim Ermolayev¹ and Vyacheslav Tolok²

¹ Department of IT, Zaporozhye National University,
66 Zhukovskogo st., 69600 Zaporozhye, Ukraine
m.davidovsky@gmail.com, vadim@ermolayev.com

² Department of Mathematical Modeling, Zaporozhye National University,
66 Zhukovskogo st., 69600 Zaporozhye, Ukraine
vyacheslav-tolok@yandex.ru

Abstract. The paper presents the results of evaluating the software solution for ontology instance migration problem in the use case involving the ontologies used in construction industry – freeClass and eClassOWL with the Bau-DataWeb dataset representing the individuals. Ontology instance migration problem is understood as a sub-problem of ontology alignment. Our methodology assumes (semi-) automated iterative process possibly involving a human for validating the results. The process consists of the two steps: (1) schema-based mappings discovery done by the agent-based matcher software; and (2) ontology instance transformation and migration according to the discovered mappings done by the ontology instance migration engine software. The evaluation experiment has been conducted in two phases and yielded results of acceptable quality in terms of precision, recall, and f-measure.

Keywords. Ontology alignment, industrial application, ontology instance migration, evaluation experiment.

Key terms. Industry, Integration, Interoperability, KnowledgeManagement-Process, AgentBasedSystem,

1 Introduction

Ontologies are being widely adopted today in the academic world and increasingly attract the attention of researchers and practitioners in information technology and knowledge-based system development and applications. Many authors, e.g. [1], argue that ontologies constitute the substance of the advanced technologies for solving the problems of interoperability, communication, and cooperation between different applications within the same environment. Indeed, ontologies conceptualize semantics of the domains within a discourse that are common for interoperating systems. Thus,

ontologies serve as a bridge for “understanding” between the systems or their parts. Despite that, application of ontologies in industry still faces several problems.

The first group of problems concerns the inertia that is typical for the process of application of advanced technologies in industry, e. g. [2]. This paper reflects the views of the practitioners who have witnessed incomprehension and opposition in trying to solve customer problems using ontologies. These problems are attempted to be resolved through establishing a closer contact with domain knowledge stakeholders and their more active involvement in the development of ontologies – e.g. [3]. Another complementary and important activity is lowering the effort for developing ontologies which could be done via providing the tool support for domain experts taking part in ontology development.

The other important stratum of problems in the application of ontologies in industry is related to the re-use of existing large industrial knowledge bases, collections, or ontologies and the exploitation of those knowledge assets within large enterprise information systems (IS). Obviously it is obligatory to provide stable interoperation of ISs in industrial settings to prevent substantial errors in maintenance, production, and sales. However the use of ontologies per se doesn't completely solve interoperability issues as it essentially raises heterogeneity problems to a higher level [4]. So, the methods for aligning ontologies need to be provided to understand and explicitly specify semantic mappings between these different conceptualisations. Industrial ontologies as a rule contain large quantity of individuals (or instances). Hence, an important and typical sub-problem of ontology alignment in industrial settings is ontology instance migration that is the process of transferring instances between aligned ontologies. The numbers of the individuals in industrial knowledge bases is very often high, so their manual alignment is not feasible. Therefore it is important to provide the tools that at least partially automate the process of alignment and do that with the quality acceptable for industries. Another important aspect of the use of ontologies in industrial settings is that industrial ISs are often distributed and belong to autonomous business entities. In such settings using intelligent software agents for ontology alignment and ontology instance migration in particular becomes an attractive implementation pathway.

The remainder of the paper is organized as follows. In section 2 we provide a classification of industrial applications of ontology alignment types of problems and describe some typical use cases. Based on this classification, we analyze industrial requirements to ontology alignment solutions. Section 3 outlines our software solution for ontology instance migration problem. Section 4 reports about the setup and results of our evaluation experiments. Finally the conclusion is given and the plans for the future work are outlined.

2 Related Work, Applications, and Use Cases

Surveys of ontology alignment for a wide range of applications can be found in [5], [6], [7]. Applications of agent-based ontology alignment and respective requirements

are analyzed in [8]. This paper focuses on industrial applications of ontology alignment in broad and ontology instance migration as its sub-problem [8].

The following industrial application categories may be outlined that require ontology alignment and instance migration solutions.

1. Industrial knowledge-driven simulation models. Simulation models are widely used in industry ([9], [10], [11]). The complexity level of modern simulation systems requires the use of knowledge-based models. This knowledge may be related to various branches of science, engineering disciplines, can contain different models satisfying different demands. This requires the use of ontologies and related activities such as ontology merging and alignment.

2. Industrial information systems in the context of Semantic Web and eCommerce. eCommerce is a type of industry where buying and selling of product or service is conducted over electronic systems such as the Internet and other computer networks. In order to perform such an exchange of business information, this information must contain product (or service) descriptions. As a rule such information is presented in the form of product or service ontology [12]. Good examples of such ontologies are [13], [14], [15]. When a business process involves more than one party or in a case of using more than one source respective ontologies obviously have to be aligned. This situation is also typical for The Semantic Web where ontologies along with intelligent software agents are the main pillars [16].

3. Integration and interoperability of heterogeneous enterprise ISs. Today information ecosystem of a modern enterprise as a rule contains numbers of applications from different vendors and used for different purposes. In order to effectively use these heterogeneous applications together with distributed data and knowledge repositories they must be integrated into a single system. Likewise implementation and deployment of new software solutions must be reconciled and integrated with legacy software systems. Here ontologies may be used not only as domain knowledge representation models, but also as mediators for integration of heterogeneous applications. Enterprise integration attracts substantial interest of research community and a number of solutions are proposed (e.g., [17], [18]).

4. Knowledge sharing and migration between enterprise ISs. Interaction and cooperation of modern enterprises often implies knowledge sharing and migration. In such a way enterprise may enrich and harmonize their knowledge assets. In this case knowledge models obviously must be reconciled and aligned. This issue is not widely addressed in literature (but some early efforts, e.g. [19], are described) as it usually requires some (combination of) typical ontology management activities (such as ontology evolution and knowledge sharing – please see some details above).

Each of the application categories sets up some requirements to specific alignment methods used within the category. Due to the wide variety of ontologies used in industry it is difficult to set up a detailed set of requirements for ontology matching methods. These requirements may substantially vary depending on ontology size and structure so we outline only the most general observations. We analyze the requirements for ontology alignment regardless to industrial application in [8].

Run-time. 1st and 2nd categories assumed the matching process to be performed at run-time. In that case the maximum level of automation must be reached. In 3rd

and 4th categories it is allowed to perform matching and relative activities previously and separately. This allows active involvement of experts to the matching process (for alignment validation, relevance verification, etc.).

Completeness. Completeness is of the most importance in the 1st and 3rd cases. It is important not to miss knowledge in these cases. At the same time, in the 2nd category the response time of method implementation to a system query is more critical as in that case matching is usually performed during runtime.

Relevance. In the 4th case, the relevance of knowledge is the most critical (particularly during migration from an older system to a newer one). Here it is first of all important to save actual knowledge, but some obsolete knowledge may be discarded.

3 Solution Overview

The main focus of the paper is evaluation of ontology alignment and instance migration methodology in industrial settings. The methodology assumes (semi-) automated iterative process of ontology alignment and instance migration with possible human intervention for checking the correctness and setting up the process. The overall methodology consists of two steps: (1) mapping discovery and determination of structural differences between ontologies and (2) ontology instance transformation and migration according to the determined differences.

The first step is essentially the process of ontology matching with the only difference that it results not only in ontology alignment but also produces an output of a set of transformation rules that further drive the process of ontology instance migration. The solution for the first step is based on the implementation of meaning negotiation between intelligent agents (we call this agent-based solution ABOA matcher [8]). The matching process embodies the strategy that originates from [20] and is described in detail in [21]. Negotiations among the agents are conducted in an iterative way and with an aim to reduce the semantic distance between the negotiated structural contexts of the respective ontology schemas. A negotiation is stopped when the distance reaches a commonly accepted threshold or the parties exhaust their propositions and arguments.

At the second step agents use Instance Migration Engine in order to transfer instances between ontologies based on the transformation rules generated at the first step. Instance migration results in the transfer of all the assertions that do not require the resolution of the problem cases by the ontology engineer. The cases that caused problems are recorded in the migration log. The details on the second step of the methodology are described in [22].

4 Evaluation Experiment

To test our methodology and solution of ontology alignment and instance migration we choose real industrial ontologies: freeClass¹ ontology for construction and building materials and services and eClassOWL² [14] – the web ontology for products and services. The dataset of the European building and construction materials market for the Semantic Web (BauDataWeb³) has been selected as the set of assertions for migration. Structural parameters of the ontologies are presented in Table 1. General experimental set-up specified in ISO/IEC 24744 notation for describing methodologies [23] is pictured in Figure 1.

Table 1. Structural parameters of industrial ontologies used in the second experiment

	Total number of axioms	Number of logical axioms	Number of classes	Number of object properties	Number of data properties	Number of individuals
freeClass	78414	9622	5231	168	3	1335
eClassOWL	360243	117090	60662	4900	2453	4766
BauDataWeb	-	-	-	-	-	Over 60 million instances

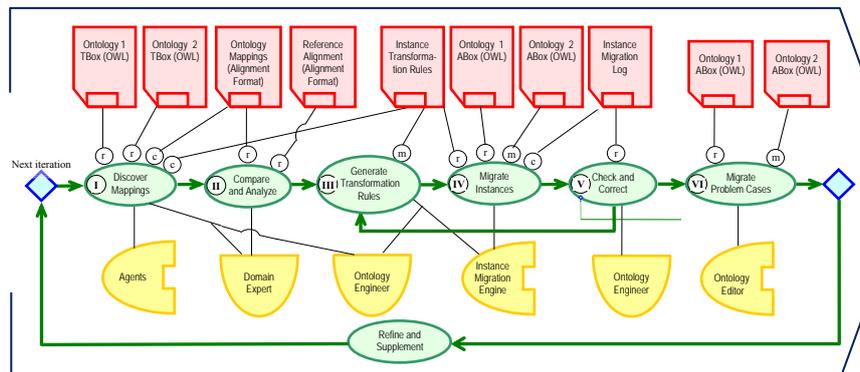


Fig. 1. The set-up of the evaluation experiment

The test case doesn't contain any reference alignment. Hence, we had to determine reference mappings manually in order to objectively judge about the obtained results. For convenience both freeClass and eClassOWL ontologies may be divided into 2 parts. The first parts are actually the sets of entities directly inherited from the

¹ <http://www.freeclass.eu/> – the ontology for construction and building materials and services

² <http://www.heppnetz.de/projects/eclassowl/> – the web ontology for products and services

³ <http://semantic.eurobau.com/> – BauDataWeb: the European Building and Construction Materials Database for the Semantic Web

GoodRelations ontology⁴ [13] and also some concepts from other common-sense vocabularies. The schemas of those parts of the ontologies are almost identical, so the difference is mostly in the sets of individual assertions. Further, those parts do not cause any problems in the discovery of the reference mappings as the entities mainly have human-understandable names and labels. Based on the analysis of the above-mentioned parts of the ontologies we constructed the set of reference mappings (further mentioned as Alignment 1). The second parts of the ontologies consist of internal entities that do not have understandable names (the names represent some identifiers composed of numbers and characters), but some of them still have labels with descriptions. Due to the big quantity of those entities we did not analyze the whole sets and choose the 20 entities that are semantically similar. Then we discovered respective mappings for those chosen entities (further mentioned as Alignment 2). The parameters for both alignments are presented in Table 2 where for brevity we include only the information about the classes and properties.

Table 2. Parameters of reference alignment for the experiment with the BauDataWeb dataset

	Number of mapped entities		
	Classes	Object properties	Datatype Properties
Alignment 1	53	55	53
Alignment 2	20	11	0

Thus, the experiment with the BauDataWeb dataset has been performed in two phases. Within the first phase we constructed the reference alignment (Alignment 1) and started the matching process using the ABOA matcher. Then we found the mappings that correspond to Alignment 1 and compared them to the reference ones. Alignment quality values for the results of this step are very high (Table 3, row 1) as these parts are almost identical.

Table 3. Matching results

Experiment Step	Alignment Quality Measures		
	Precision	Recall	F-Measure
1	0.99999	0.99999	0.99999
2	0.69552	0.42384	0.52671

It might be considered that the Alignment 1 in our experiment is not a topically interesting case as the semantic differences are tiny and could be easily discovered manually. However, this experimental phase represents a good case for validating the generated instance transformation rules and instance migration quality. In this phase all of the generated transformation rules were correct. More details on the transformation rules could be seen in [22]. Within the second phase we determined the Alignment 2 and tried to find the respective mappings within the alignment discovered by the matcher. The alignment quality measures for the second phase are lower than for

⁴ <http://www.heppnetz.de/projects/goodrelations/> – the web vocabulary for e-commerce

the phase 1, which is conditioned by the relatively weak semantic similarity between the structural contexts [20] that correspond to these parts of ontologies. It is also worth noticing that the Precision value is noticeably higher than the Recall one within phase 2. It is so because string-based structural similarity measurement methods yield high values on labels. Labels can contain parts (e.g. words) that are common for many of them, but respective entities in general are not semantically similar. For example the comparison of labels “construction technology” and “pump technology” will give noticeably high similarity values. However those labels belong to the entities that are obviously not that similar semantically.

5 Concluding Remarks and Future Work

The paper presented the experiment evaluating our methodology and software solution for ontology instance migration on real-world industrial ontologies. The experiment shows acceptable results that allow a positive judgement about the applicability of our methodology in industrial settings. The results also suggest some directions for the future work. The experiment with large ontologies (BauDataWeb dataset) shows that the ontology instance migration engine allows migrating about several million instances using a conventional desktop computer. Hence, a technique to overcome this upper limit is needed for scaling the tool up to the volumes characteristic to Big Semantic Data. Looking for such a technique is on our research and development agenda. In the future we also plan to conduct a series of experiments with the ontologies specified in OWL sublanguages⁵ and OWL 2 profiles⁶. Another important direction for the future research is evaluating our approach on ontologies having different structural patterns like a taxonomy (tree-type) structure, a network structure (ontologies rich with object properties), OWL graphs with high and low vertex degrees, etc.

Acknowledgments

The authors are grateful to the colleagues who provided the industrial ontologies for our experiments – Univ. Prof. Dr. Martin Hepp and Dipl. Ing. Andreas Radinger from E-Business and Web Science Research Group of the Universität der Bundeswehr, München.

References

1. Bittner, T., Donnelly, M., Winter, S.: Ontology and Semantic Interoperability. In: D. Proserpi and S. Zlatanova (ed.) Large-scale 3D Data Integration: Problems and Challenges. London, CRCPress (2005)

⁵ <http://www.w3.org/TR/2004/REC-owl-features-20040210/#s1.3>

⁶ <http://www.w3.org/TR/owl2-profiles/>

2. Malzahn, D.: Industrial Application of Ontologies. In: eKNOW 2011, The Third International Conference on Information, Process, and Knowledge Management, (2011)
3. Tatarintseva, O., Ermolayev, V., Fensel, A.: Is Your Ontology a Burden or a Gem? – Towards Xtreme Ontology Engineering. In: Ermolayev, V. et al. (eds.) Proc. ICTERI 2011, CEUR-WS.org/Vol-716, 65-81 (2011)
4. Euzenat J., Shvaiko P.: *Ontology Matching*. Berlin Heidelberg, Springer-Verlag (2007)
5. Corcho, O.: Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & Knowledge Engineering* 46, 41–64 (2003)
6. Ehrig, M.: *Ontology Alignment: Bridging the Semantic Gap (Semantic Web and Beyond)*. Springer (2006)
7. Zhdanova A. V., de Bruijn, J., Zimmermann, K., Scharffe, F.: *Ontology Alignment Solution*. Deliverable D14 v2.0, (2004)
8. Ermolayev, V., Davidovsky, M.: Agent-Based Ontology Alignment: Basics, Applications, Theoretical Foundations, and Demonstration. Tutorial Paper. In: Dan Burdescu, D., Akerkar, R., Badica, C. (eds.) Proc. WIMS 2012, 11-22, ACM (2012)
9. Silver, G., Hassan, O.H., Miller, J.: From domain ontologies to modeling ontologies to executable simulation models. In: Proc. of the 2007 Winter Simulation Conference, (2007)
10. Novák, P., Šindelář, R.: Applications of ontologies for assembling simulation models of industrial systems. In: Proc. of the 2011th Confederated international conference on the move to meaningful internet systems (OTM'11), pp.148–157, Springer-Verlag Berlin, Heidelberg (2011)
11. Ermolayev, V., Keberle, N., Matzke, W.-E.: An Upper Level Ontological Model for Engineering Design Performance Domain, LNBIP, vol. 20, pp.127–141. Springer, Heidelberg (2008)
12. Ding, Y., Fensel, D., Klein, M., Omelayenko, B., Schulten, E.: The Role of Ontologies in eCommerce. In: Steffen Staab, Rudi Studer (eds.): *Handbook on Ontologies*. International Handbooks on Information Systems. pp. 593-616, ISBN 3-540-40834-7, Springer (2004)
13. Hepp, M. GoodRelations: An Ontology for Describing Products and Services Offers on the Web, LNCS, vol. 5268, pp. 332–347. Springer Berlin Heidelberg (2008)
14. Hepp, M.: Products and Services Ontologies: A Methodology for Deriving OWL Ontologies from Industrial Categorization Standards. In: *Int'l Journal on Semantic Web & Information Systems* 2(1) (2006), pp. 72–99, (2006)
15. Morgenstern, L., Riecken, D.: SNAP: An Action-Based Ontology for E-commerce Reasoning. In: Proc., *Formal Ontologies Meet Industry*, Verona, Italy, (2005)
16. Berners-Lee, T., Hendler, J., Lassila, O.: *The Semantic Web* (Berners-Lee et. al 2001). *Scientific American* 284, 28–37 (2001)
17. Izza, S., Vincent, L., Burlat, P.: A Unified Framework for Enterprise Integration: An Ontology-Driven Service-Oriented Approach. In: Pre-proc. of the First International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA'2005), pp. 78–89. Geneva, Switzerland, February 23 – 25, (2005)
18. Stoutenburg, S. et al.: Ontologies in OWL for Rapid Enterprise Integration. *Time* 122, 82–89 (1994)
19. Lochovsky, F. H., Woo, C. C., Williams, L. J.: A micro-organizational model for supporting knowledge migration. In: Proc. of the ACM SIGOIS and IEEE CS TC-OA conference on Office information systems, pp.194–204. Cambridge, Massachusetts, US, (1990)
20. Ermolayev V., Keberle, N., Matzke, W.-E., Vladimirov, V.: A Strategy for Automated Meaning Negotiation in Distributed Information Retrieval. In: Y. Gil et al. (Eds.): *ISWC 2005 Proc. 4th Int. Semantic Web Conference (ISWC'05)*, 6–10 November, Galway, Ireland. LNCS 3729, pp. 201–215 (2005)

21. Davidovsky, M., Ermolayev, V., Tolok, V.: Agent-based implementation for the discovery of structural difference in OWL DL ontologies. In: Mayr, H. C., Ginige, A., Liddle, S. (ed.) Proc. Fourth Int United Information Systems Conference (UNISCON 2012). LNBIP 137, Berlin, Heidelberg: Springer-Verlag (2013)
22. Davidovsky, M., Ermolayev, V., Tolok, V.: Instance Migration between Ontologies having Structural Differences. *Int. J. on Art. Int. Tools.* 20(6), 1127-1156 (2011)
23. Henderson-Sellers, B., Gonzalez-Perez, C.: Standardizing Methodology Metamodelling and Notation: An ISO Exemplar. In: Kaschek, R., Kop, C., Steinberger, C., Fliedl, G. (eds.) UNISCON 2008. LNBIP, vol. 5, pp. 1–12. Springer, Berlin/Heidelberg, (2008)

Extracting Knowledge Tokens from Text Streams

Eugene Alferov^{1,2} and Vadim Ermolayev¹

¹ Department of IT, Zaporozhye National University,
66 Zhukovskogo st., 69063, Zaporozhye, Ukraine

alferov.evgeniy@gmail.com, vadim@ermolayev.com

² Kherson State University, 27, 40 Rokiv Zhovnya ave., 73000, Ukraine

alferov_jk@ksu.ks.ua

Abstract. This problem analysis paper presents our position on how could the solution be sought to the problem of extracting semantically rich fragments from a stream of plain text posts. We first present our understanding of the problem context and explain the focus of our research. Further, in the problem setting section we elaborate the workflow for knowledge extraction from incoming information tokens. This workflow is then used as a key to structure our review of the literature on the relevant component techniques which may be exploited in a combination to achieve the desired outcome. We finally outline our plan for conducting the experiments with an aim to validate the workflow and find a proper combination of the component techniques for all steps which may solve our specific research problem.

Keywords. Workflow, knowledge extraction, text streams, processing, ontology learning, component techniques

Key terms. Data, Process, Knowledge, Approach, Methodology

1 Introduction

The dramatic growth of data volumes we face today is accelerated by the increase of social networking applications that allow non-specialist users create a huge amount of content easily and freely. Equipped with rapidly evolving mobile devices, a user is becoming a nomadic gateway boosting the generation of additional real-time sensor data. The emerging Internet of Things makes each and every thing a data or content, adding billions of additional artificial and autonomic sources of data to the overall landscape. Smart spaces, where people, devices, and their infrastructures are all loosely connected, also generate data of unprecedented volumes and with velocities rarely observed before. Noticeably, the major part of the new data comes in streams.

An expectation is that valuable information will be extracted out of all these data to help improve the quality of life and making our world a better place – for humans.

Humans are however left bewildered about how to use, analyze, understand all these data, giving a proper account to its dynamics. A topical recent estimate of the need for data-savvy managers in the United States is 1.5 million [1]. This manpower is needed to extract and use valuable information and knowledge for further decision making. The critical steps in this work are (i) extracting information and knowledge; and (ii) bringing the descriptions of the reflections of the world or domain into a refined state – accounting for the changes brought in by new data, at scale.

In this paper we focus on the step (i) extraction. In Section 2 we present the problem statement by giving basic definitions and providing our view on how could a processing workflow look like. The plethora of approaches, techniques, technologies, and software tools already exist for solving different parts of the overall problem. Hence we analyze the related work and structure this analysis using the workflow as the key in Section 3. Finally we conclude the paper and present our plans for the future proof of concept experimental work in Section 4.

2 Problem Statement

Ontology is a complex artifact that comprises structural components of several types. Further the structural denotation of an ontology used in Description Logics [2] is exploited: an ontology O comprises its schema S and the set of individuals I : $O = (S, I)$. Ontology schema is also referred to as a terminological component (TBox). It contains the statements describing the concepts of O , the properties of those concepts, and the axioms over the schema constituents.

If a finer grained look at an ontology schema is taken, one may consider S comprising the following interrelated constituents: $S = \{S^C, S^O, S^D, S^A\}$, where S^C is the set of statements describing concepts, S^O is the set of statements describing object properties, S^D is the set of statements describing datatype properties, and S^A is the set of axioms specifying constraints over S^C , S^O , and S^D (c.f. [3]). One may notice that these constituents correspond to the types of the schema specification statements of an ontology representation language L which is used for specifying O .

The set of individuals, also referred to as assertional component (ABox), is the set of the ground statements about the individuals and their attribution to the constituents of the schema.

Ontology Learning is the process of extracting the abovementioned constituents of O from a text stream source. More specifically, the problem which is approached in this research work is twofold:

For every individual plain text document (further referred to as information token) arriving in the stream window DO:

- (i) Extract ontological fragment (further referred to as knowledge token) specifying the semantics of the information token.
- (ii) Refine the ontology O incorporating the changes brought in by the knowledge token.

The focus of this paper is the first part of the problem – the extraction of knowledge tokens from information tokens of plain text in a particular professional domain coming in a stream. The texts of ICTERI paper abstracts have been chosen as the domain and source text corpus for our initial experiments – see also Section 4.

As an ontology is a complex artifact, the extraction of knowledge tokens from texts is also a complex process. It comprises several steps and, possibly, iterations for extracting different structural constituents of S and I . These steps produce several types of outputs in a particular sequence, sometimes referred to as the ontology learning layer cake (c.f. [4]). Those outputs are terms – concepts and their instances – datatype properties – taxonomic relationships and object properties – axioms. Based on [5] we present in Fig. 1 a workflow putting together extraction steps, inputs, outputs, and required component technology types.

The overall workflow contains two consecutive phases – Text Pre-processing and Ontology Extraction. Text Pre-processing phase gets the information token as a plain text input and produces its structured representation as a set of terms by applying several statistical and linguistic techniques. All the tasks of the Ontology Extraction Phase use the output of Phase 1 as their input and incrementally build up the knowledge token by adding different ABox and TBox constituents. For that statistical, linguistic, semantic, and logical techniques are employed in combinations. Fig. 1 lists all relevant component techniques per task. All of those are never used in implementations. Therefore our initial research objective is to find out which combination of component techniques works best of all for our specific data – i.e. copes well with (a) the texts of small size but belonging to a particular domain; and (b) limited processing time constrained by a stream window lifetime parameter. Further, after this constellation of component techniques is chosen, the objective would be to refine those which do not provide results of a satisfactory quality in our problem settings.

3 Related Research and Available Component Techniques

In this section we will describe the component techniques, outlined in Fig. 1, which we found relevant to our work. Those component techniques could overall be categorized as linguistic, statistic, semantic and logical (c.f. [5]). As pictured in Fig. 1 they could be applied at different steps and for different purposes. Though not explicitly shown in Fig. 1, the steps may undergo iterations for refining their results. Therefore, the workflow proposed in this paper could be considered as hybrid and iterative.

De-noising (statistical, linguistic). This is a method that extracts the de-noised text, comprising the content-rich sentences, from full texts [6]. Processing of noisy text becomes important because the quality of texts in the form of blogs, emails and chat logs can be extremely poor. The sentences in dirty texts are typically full of spelling errors, ad-hoc abbreviations and improper casing [7].

Tokenization. Tokenization is splitting the text into a set of tokens, usually words. This process is unsupervised and can be performed automatically by program-parser.

Part of speech detection/tagging (linguistic). Part of speech tagging (POST) is the process of assigning one of the parts of speech to the given word. POST provides the

syntactic structures and dependency information required for further linguistic analysis in order to uncover terms and relations. POST is a semi-supervised or even unsupervised process.

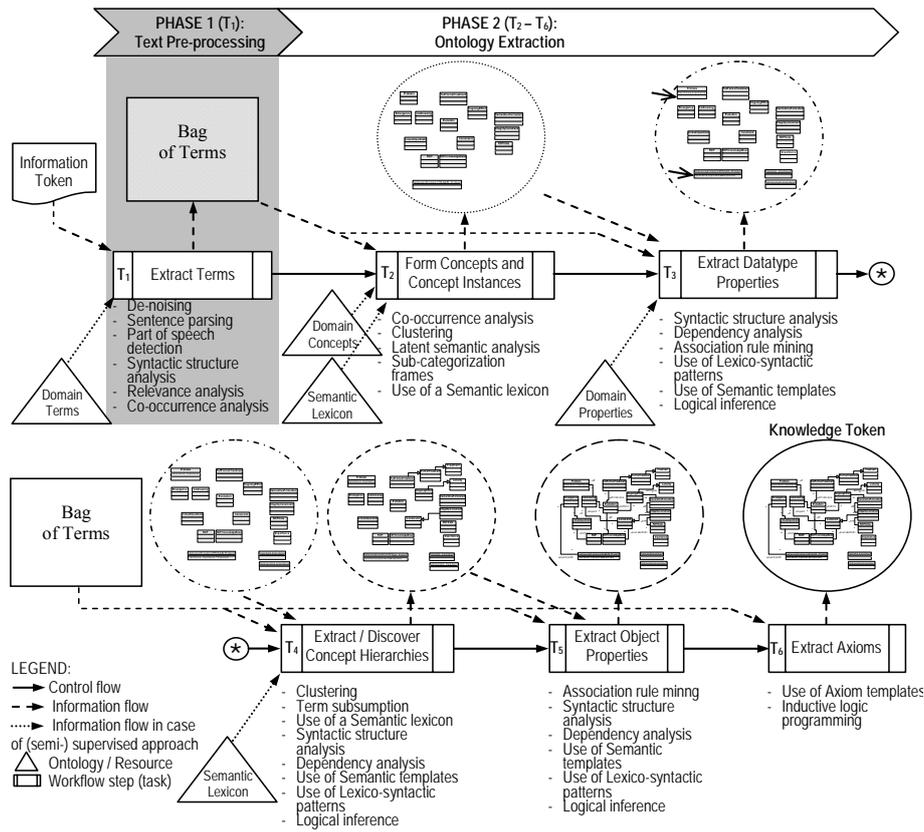


Fig. 1. A workflow for knowledge token extraction

Lemmatization (linguistic). Lemmatization is the reduction of morphological variants of the tokens to their base form that can be performed in unsupervised way. For achieving this word form must be known, i.e. the part of speech of every word has to be assigned in the text document. This process usually takes a time and may contain errors.

Chunking (linguistic). Chunking is unsupervised splitting a text in syntactically correlated parts.

Sentence parsing. Sentence parsing is identifying the syntactic structure of a sentence, for example in a form of a parse tree.

Syntactic structure analysis (linguistic). In syntactic structure analysis, words and modifiers in syntactic structures (e.g., noun phrases, verb phrases, and prepositional phrases) are analyzed to discover potential terms and relations. It can be done in unsupervised way.

Relevance Analysis (statistical). The extent of occurrence of terms in individual documents and in text corpora is employed for relevance analysis. This is semi-supervised or even unsupervised technique.

Co-occurrence analysis (statistical). Co-occurrence analysis identifies lexical units that tend to occur together for purposes ranging from extracting related terms to discovering implicit relations between concepts [5]. This technique is unsupervised.

Clustering (statistical). Grouping together variants of terms to form concepts and separating unrelated ones is known as terms clustering. It usually unsupervised technique. In this approach some measure of similarity is employed to assign terms into groups for discovering concepts or constructing hierarchy [8]. Some of the major issues in clustering are working with high-dimensional data and feature extraction and preparation for similarity measurement. This gave rise to a class of featureless similarity measures based solely on the co-occurrence of words in large text corpora. It is known that clustering results are of acceptable quality only if a statistically representative (i.e. large) text corpora is processed. This fact limits the applicability of this technique in our settings (texts of small size). However, used in the combination with other techniques, clustering may yield some valuable addition to the result – and thus needs to be tried.

Latent semantic analysis (statistical). Latent semantic analysis (LSA) is a theoretical approach and mathematical method for determining the meaning similarity of words and passages by analysis of large text corpora. The main idea is that the aggregate of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other [9]. LSA can be useful in our investigation because it is a fully automatic mathematical and statistical technique for extracting and inferring meaningful relations from the contextual usage of words in text.

Sub-categorization (linguistic, semantic). Sub-categorization, or extracting sub-categorization frames, is an approach to extract one type of lexical information with particular importance for Natural Language Processing (NLP). Access to an accurate and comprehensive sub-categorization lexicon is vital for the development of successful parsing technology important for many NLP tasks (e.g. automatic verb classification) and useful for any application which can benefit from information about predicate-argument structure (e.g. Information Extraction) [10].

Using semantic lexicon (linguistic, semantic). A semantic lexicon is a dictionary or thesaurus of words/terms labeled with semantic classes (e.g., “ongoing effort” is an Activity) so associations can be drawn between words that have not previously been encountered [11]. Semantic lexicons are a popular resource in ontology learning and play an important role in many NLP tasks.

Dependency analysis (linguistic). Syntactic structure consists of lexical items, linked by dependencies. They are binary asymmetric relations that are held between a head and its dependents. Dependency analysis examines dependency information to uncover relations at the sentence level. In this analysis, grammatical relations, such as subject, object, adjunct, and complement, are used for determining more complex relations. Dependency analysis is usually unsupervised approach.

Association rule mining (statistical). Association rule mining aims to extract correlations, frequent patterns, associations or casual structures among sets of items in data repositories [12]. It is an unsupervised component technique which works well for considerably big data corpora. Association rules highlight correlations between features in the texts, e.g. keywords. Association rules can be easily interpreted and are understandable for an analyst or even for a normal user.

Use of lexico-syntactic patterns (linguistic). Lexico-syntactic patterns (LSPs) are generalized linguistic structures or schemas that indicate semantic relationships among terms and can be applied to the identification of formalized concepts and conceptual relations in natural language text [13]. Lexico-syntactic patterns are suitable for automatic ontology building, since they model semantic relations. These display exactly the kind of relation between their parts that makes them easily translatable into an ontology representation.

Use of semantic templates (semantic, linguistic). Semantic templates are similar to lexico-syntactic patterns in terms of their purpose. However, semantic templates offer more detailed rules and conditions for extracting not only taxonomic relations but also complex non-taxonomic relations [5].

Logical inference (logical, semantic). In logical inference implicit relations are derived from existing ones using rules such as transitivity and inheritance [5]. However, the introduction of invalid or conflicting relations may also happen in case of an incomplete or underspecified inference rule set – for example because of improper account for the validity of transitivity or mutual disjointness axioms.

Term subsumption (statistical, semantic). In the subsumption method, a given term subsumes another term if the documents in which the latter term occurs are a subset of the documents in which the given term occurs [14]. A term subsumption measure is used to quantify the extent of a term x being more general than another term y . This technique is semi-supervised and unsupervised too. The term subsumption technique is easy to implement and it makes labeling concepts an easy task. However, with this method, it is difficult to classify terms that do not co-occur frequently and it requires a large data set to work reliably.

Use of axiom templates (semantic, linguistic). Axioms are useful for describing the relationships between the concepts of an ontology. They can be written in different ways depending on the relation that exist among the concepts.

Inductive logic programming (logical, semantic). Inductive logic programming (ILP) is a research area at the intersection of inductive machine learning and logic programming. ILP generalizes the inductive and the deductive approaches by aiming to develop theories, techniques and applications of inductive learning from observations and background knowledge represented in first order logical framework.

The overview of the applicability of the presented component techniques and their interrelationship with respect to the tasks in our workflow are presented in Table 1.

4 Summary and Future Work

Our literature search has revealed that extracting knowledge, or more specifically learning ontologies, from plain text corpora is a well developed research field that continues to produce new results. However, and to the best of our knowledge, extracting ontologies from text streams, with a constraint on the life time of an input information token, is a recently emerged research problem. The reasons for adding this specific problem to the research agenda are the phenomenon of Big Data, in particular its velocity dimension, as well as the need for better, more reliable, semantically rich solutions for automating Big Data analytics. One more complication introduced by our problem setting is the small size of an individual information token which hinders yielding good quality results using the majority of traditional statistical and linguistic techniques for ontology extraction from text corpora.

We argued in this paper that applying a combination of the relevant existing component techniques in a structured and iterative way may overall produce such a result – as an incremental collection of ontology elements in a knowledge token provided by individual techniques at different stages in our proposed workflow.

Table 1. Relevance of component techniques to the tasks within the workflow for extracting knowledge tokens from information tokens

Component technology	Task (Fig. 1.)					
	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆
De-noising	st, li					
Part of speech detection/tagging	li					
Lemmatization	li					
Chunking	li					
Syntactic structure analysis	li		li	li	li	
Relevance Analysis	st					
Co-occurrence analysis	st	st				
Clustering		st		st		
Latent semantic analysis		st				
Sub-categorization		se, li				
Using semantic lexicon		se, li		se, li	se, li	
Dependency analysis			li	li	li	
Association rule mining			st		st	
Use of lexico-syntactic patterns			li	li	li	
Use of semantic templates			se, li	se, li		
Logical inference			lo, se	lo, se	lo, se	
Term subsumption				st, se		
Use of axiom templates						se, li
Inductive logic programming						lo, se

Legend : li – linguistic; lo – logical; se – semantic; st – statistical;

As this research is in an early phase, we do not yet have the proof for this hypothesis. However there is the plan in place for conducting the initial series of the “proof-of-concept” experiments in which the component technologies will be exploited in a semi-supervised or supervised fashion. For that we plan to use a small but well se-

manically annotated corpus of the abstracts (information tokens) and full texts of ICTERI papers collected in the ICTERiWiki portal¹. This document corpus is incrementally extended by adding the papers and their semantic annotations for each new ICTERI conference instance. The annotations are done using the ICTERI Scope Ontology by Tatarintseva et.al. [15]. These annotations will be used as a “Golden Standard” for evaluating the results of automated knowledge token extraction using the workflow proposed in this paper.

After the concept is proven and the constellation of the component techniques is circumscribed, we plan to test the approach on one of the professional news portals. Further, it is planned to extend the proposed knowledge extraction procedure to sensor stream data processing.

References

1. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Hung Byers, A.: Big data: the Next Frontier for Innovation, Competition, and Productivity. McKinsey Global Institute (2011), http://www.mckinsey.com/insights/mgi/research/technology_and_innovation/big_data_the_next_frontier_for_innovation
2. Nardi, D., Brachman, R.J.: An Introduction to Description Logics. In: Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., Patel-Schneider, P. F. (eds.) *The Description Logic Handbook*, Cambridge University Press New York, NY, USA (2007)
3. Davidovsky, M., Ermolayev, V., Tolok V.: Instance Migration between Ontologies Having Structural Differences. In: *Int. J. on Artificial Intelligence Tools*, vol. 20(6), pp. 1127–1156 (2011)
4. Buitelaar, P., Cimiano, P., Magnini, B.: Ontology Learning from Text: an Overview. In: Buitelaar, P., Cimiano, P., Magnini, B. (eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*, IOS Press, Amsterdam (2005)
5. Wong, W., Liu, W., Bennamoun, M.: Ontology Learning from Text: a Look Back and into the Future. *ACM Comput. Surv.*, 44(4), Article 20, 36 pages. <http://doi.acm.org/10.1145/2333112.2333115> (2012)
6. Shams, R., Mercer, R. E.: Investigating Keyphrase Indexing with Text Denoising. In: *Proceedings of the 12th ACM/IEEE-CS Joint Conf. on Digital Libraries*, pp. 263–266, ACM (2012)
7. Wong, W., Liu, W., Bennamoun, M.: Enhanced Integrated Scoring for Cleaning Dirty Texts. arXiv preprint arXiv:0810.0332. (2008)
8. Cimiano, P., Hotho, A., Staab, S.: Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. *Journal of Artificial Intelligence Research Archive*, 24(1), 305–339 (2005)
9. Landauer, T.K., Foltz, P.W., Laham, D.: Introduction to Latent Semantic Analysis. *Journal: Discourse Processes*, 25(2-3), 259–284 (1998)
10. Preiss, J., Briscoe, T., Korhonen, A.: A System for Large-Scale Acquisition of Verbal, Nominal and Adjectival Subcategorization Frames from Corpora. In: *Annual Meeting. Association for Computational Linguistics*, 45(1), 912 (2007)
11. Thelen, M., Riloff, E.: A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts. In: *Proc. ACL-02 Conf. on Empirical Methods in Natural*

¹ <http://isrg.kit.znu.edu.ua/icteriwiki/>

- Language Processing, Association for Computational Linguistics, vol. 10, pp. 214–221 (2002)
12. Kotsiantis, S., Kanellopoulos, D.: Association Rules Mining: a Recent Overview. *GESTS International Transactions on Computer Science and Engineering*, 32(1), 71–82 (2006)
 13. Summary on Requirements on Lexico-Syntactic Patterns (Synthesis by PC), http://www.w3.org/community/ontolex/wiki/Specification_of_Requirements/Lexico-Syntactic_Patterns
 14. De Knijff, J., Frasincar, F., Hogenboom, F.: Domain Taxonomy Learning from Text: the Subsumption Method versus Hierarchical Clustering. *Data & Knowledge Engineering*, (2012)
 15. Tatarintseva, O., Borue, Yu., Ermolayev, V.: Validating OntoElect Methodology in Refining ICTERI Scope Ontology. In: H.C. Mayr et al. (Eds.): *UNISCON 2012, LNBIP 137*, pp. 128–139 (2013)

1.3 Model-Based

Software System Development

Use of Neural Networks for Monitoring Beam Spectrum of Industrial Electron Accelerators

Oleksandr Baiev, Valentine Lazurik and Ievgen Didenko

School of Computer Science, V. N. Karazin Kharkiv National University,
4, Svobody Sqr., 61022, Kharkiv, Ukraine

oleksandr.baiev@gmail.com, lazurik@hotmail.com,
ievgen.v.didenko@gmail.com

Abstract. This paper investigates technique for solving spectrometry inverse problem the neural network as method for reconstruction of electron beam spectrum using depth-charge curve. The inverse problem turned into multivariable optimization and the form of spectrum is based on proposed three-parameter model. Radial basis function network calculates the parameters of this model. We developed computational experiment using Monte-Carlo technique to evaluate strengths and weaknesses of proposed approach and compare neural networks with conventional data evaluation methods.

Keywords. Neural nets, Inverse problems, Monte Carlo, Radiation technologies, Depth-charge curve

Key terms. ComputerSimulation, Methodology, MachineIntelligence

1 Introduction

One of the main characteristic of the irradiation processes is an energy of beam. This parameter influences on absorbed dose in target. Therefore, standards for radiation technologies [1, 2] predetermine the upper bound of beam energy to prevent ionization of the object under irradiation. Because of accelerator features, electrons in beam have different energy. Thus, the beam energy represented by some function, which shows relations between particles number and their energies. This function called beam spectrum. In practice at least three parameters define the spectrum: average (E_{av}) and probably (E_p) energies and full width on half maximum (E_w). In order to measure beam energy dosimetric wedge and stack are widely used in centers of radiation technologies. These devices allow to determine only average and probable energies of beam [1-6]. Of course, these two parameters does not allow to reconstruct full energy distribution. Thereby developing of new instruments and methods of dosimetric measurements is actual problem.

Mentioned devices intend to measure distributions of absorbed dose or charge [5, 6]. The measured depth-dose (depth-charge) curves relate to beam spectrum

through Fredholm integral equation and finding exact spectrum is an ill-posed inverse problem [7]. This means that evaluated spectrum obtained by conventional mathematical methods can differ with true energy distribution. There are, for example, method of least squares (MLS) or method of Tikhonov regularization (MTR). Above all, important disadvantage of the MLS and MTR is impossibility to include additional solution conditions, for example, correlations between parameters, positivity and other. This lack can bring to violation of conditions, given by physical laws. It should be mentioned that in common case the neural networks (NN) solve approximation tasks and find solutions based on existing precedents after supervised training [8–12]. So the one of the ways of improving dosimetry effectiveness is developing of methods for measurement results evaluation based on neural networks. In order to apply NN for dosimetric data processing it is necessary to solve next problems: select networks topology, obtaining data for NN training, developing methods for data preprocessing and interpretation, system for evaluation network effectiveness.

So current research is about feasibility of using neural networks for developing system of measurement results evaluation for beam spectrum monitoring of industrial electron accelerators. We will discuss mathematical model of measurement process, which was built in order to compile training set for network learning procedure (Section 2). Section 3 describes methods under investigation. In section 4, we will show approach for methods evaluation, which contains computational experiment and comparison criteria. In section 5 given comparison results of neural networks and conventional methods testing.

2 Physical process and mathematical model

In order to calculate radiation energy, it is a common practice in field of radiation technologies to measure depth-dose curve by dosimetric wedge. However, the works of recent years propose new devices based on measurement of depth-charge curve that can realize on-line energy monitoring [3–6]. In this work, we will consider mathematical abstraction of these devices and will build method for beam spectrum controlling using depth-charge curve.

2.1 Devices

Device [5] consists of two plates only and intend to calculate probable energy as a value which linearly depends on charge in first plate to sum charge ratio. Measurer in [6] contains 10 absorbers. But in order to simplify average energy calculation the plates were combined and authors use similar to [5] dependency.

Fig. 1 shows principal schema of measurer. Dosimetric stack consists of set of plates - absorbers. The absorbers material is often aluminum, because of radiation ruggedness. The electron beam falls on the sequence of plates. Electrons stop at different depths depending on their energy. Thus, absorbers collect some charge which can be measured by current integrators connected to corresponding plate. The set of measured values represents the depth-charge curve.

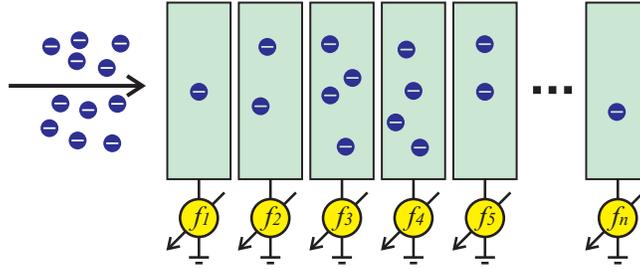


Fig. 1. Common schema of stack for depth-charge measurement

Mathematical model of the measurement process is based on a semi-empirical model of the depth-charge distribution for monoenergetic electrons and model of charge measurement uncertainty. Direct problem describes relation between known beam spectrum and depth-charge curve through equation:

$$f(x) = \int_{E_L}^{E_R} Q(x, E)y(E) dE, \quad x \in [0, x_R], \quad (1)$$

where $y(E)$ - describes relation between number of particles and their energy (electrons spectrum), $f(x)$ - describes depth distribution of charge, x_R - measurer full width, $[E_L, E_R]$ - operating energy range of accelerator, integral kernel $Q(x, E)$ corresponds to radiation type (α, β, γ) and measurer internal characteristics (including absorbers material). Works [13, 14] describe appropriate relations for monoenergetic beam and depth-charge curve.

In the research we neglect charge leakage and suppose that distance between absorbers is neglectfully small. It means that each particle from initial beam can stops in absorbers and pass through current integrator or can pass through whole device with no impact in depth-charge curve.

The measurement results of charge distribution in absorbers is set $f = \{f_1, f_2, \dots, f_n\}$ (see Fig. 1), where n - number of absorbers, f_i - integral of $f(x)$ over the depth for i -th absorber:

$$f_i = \int_{x_i}^{x_i + \Delta x} \int_{E_L}^{E_R} Q(x, E)y(E) dE dx, \quad (2)$$

where Δx - absorbers width. Equation (2) can be approximated as:

$$f_i = \frac{\Delta x}{2} \sum_j p_j^E y_j [Q(x_k + (i-1)\Delta x, E_j) + Q(x_k + i\Delta x, E_j)], \quad (3)$$

where $i = \overline{1, n}$, $j = \overline{0, m}$, $m = (E_R - E_L)/\Delta E$ - number of steps of function $y(E)$ discretization over energy axis, ΔE - step of spectrum energy discretization, y_j - value of $y(E)$ in approximation nodes, coefficient p_j^E defines method and step

of function $y(E)$ approximation. Then the measurement process can be shown as system of linear equations:

$$Ay = f \Leftrightarrow \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}, \quad (4)$$

where elements of matrix A are:

$$a_{i,j} = \frac{\Delta x}{2} p_j^E [Q(x_k + (i-1)\Delta x, E_j) + Q(x_k + i\Delta x, E_j)]. \quad (5)$$

In order to approximate $y(E)$ by method of trapezoids, coefficients p_j^E are:

$$p_j^E = \begin{cases} \Delta E/2 & j = 0 \vee j = m \\ \Delta E & otherwise \end{cases}. \quad (6)$$

It's obvious that complexity of spectrum reconstruction grows with increasing of m (dimension of vector y). In order to reduce problem the we used parameterization of $y(E)$. As mentioned above, the general practice is denoting spectrum by parameters: E_p , E_{av} , E_w . Therefore, it is reasonable to make model of the beam spectrum, which use three parameters.

2.2 Model of electrons spectrum

Fig. 2 shows geometrical interpretation of electrons spectrum model considered in the present work. The graph of spectrum consists of two part: left exponential and right linear slopes. The parameters of this model are:

- E_{max} - maximal particles energy in the beam,
- E_p - most probable energy,
- E_s - energy of 10 times decreasing of the intensity compared to E_p electrons along left slope.

In the future discussion the Π will denotes set of spectrum parameters, i.e. $\Pi = \{E_s, E_p, E_{max}\}$.

Parameters of the model correspond to characteristics of beam used in practice according to:

$$\begin{aligned} E_p &= E_p, \\ E_w &= \frac{\ln 0.5}{\ln 0.1} (E_p - E_s) + \frac{E_{max} - E_p}{2} \\ E_{av} &= E_s + \ln \left(\frac{E_{max} - E_p}{4} + \frac{0.45(E_s - E_p)}{\ln 0.1} \right) \end{aligned} \quad (7)$$

and mathematical expression for spectrum is:

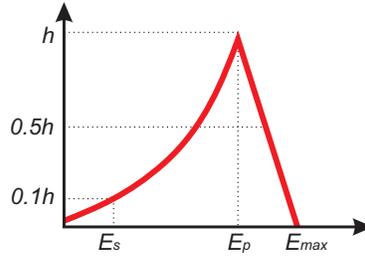


Fig. 2. Model of electron beam spectrum

$$y(E) = \begin{cases} he^{\mu(E-E_p)}, & 0 < E \leq E_p \\ k_1E + k_2, & E_p < E \leq E_{max} \\ 0, & E_{max} < E \end{cases}, \quad (8)$$

$$\mu = \frac{\ln(0.1)}{E_s - E_p}, k_1 = \frac{h}{E_p - E_{max}}, k_2 = \frac{hE_{max}}{E_{max} - E_p}, \quad (9)$$

where $E \in [0; \infty]$, $h = y(E_p)$ - maximum of function $y(E)$ and was obtained with supposition of

$$\int_{E_s}^{E_{max}} y(E) dE = 1. \quad (10)$$

Therefore, maximum of energy distribution is:

$$h = y(E_p) = [0.9 \frac{E_s - E_p}{\ln(0.1)} + 0.5(E_{max} - E_p)]^{-1}. \quad (11)$$

It should be mention, that in accordance to physical laws the function $y(E)$ is positive or equal zero for all accepted E and parameters correlates as:

$$0 < E_s < E_p \leq E_{max}. \quad (12)$$

2.3 Model of measurement

In the real experiment measured f_i differ with its real value. This error grounded on weaknesses of measurer and external influence. We will mark set of true values of $f(x)$ as f , and use \tilde{f} to mark set of values complemented with measurement uncertainty:

$$\tilde{f} = (1 + \varepsilon\xi)f, \quad (13)$$

where ε - value of standard deviation of measurement error, ξ - random variable distributed in accordance to standard normal distribution:

$$\xi = \cos(2\pi r_1) \sqrt{-2 \ln(r_2)}, \quad (14)$$

where r_1, r_2 - random variables which are distributed in accordance with standard uniform distribution.

We will use similar signature to denote evaluated parameters $\tilde{\Pi}$, \tilde{E}_s , \tilde{E}_p , and \tilde{E}_{max} reconstructed spectrum \tilde{y} instead their true values without tilde.

3 Methods for spectrum reconstruction

3.1 Neural networks

In order to apply NN for solving spectrometry inverse problem reconstruction of spectrum can be represented as multivariable function fitting. Suppose that function ϕ implements measurement process of depth-charge curve, i.e. $\tilde{f} = \phi(\Pi)$. Therefore, inverse function $\tilde{\Pi} = \phi^{-1}(\tilde{f})$ realizes transformation from depth-charge curve to beam spectrum. So approximation of ϕ^{-1} can be used to get spectrum using depth-charge curve. In the work we used general regression neural network (GRNN) [15] to fit ϕ^{-1} . This network needs set of precedence for supervised learning. Consider algorithm of training set creation.

Implemented measurement models allow to create pairs $s = (\tilde{f}, \Pi)$, where \tilde{f} calculates from parameters set Π . The collection of s is based on different Π and represents a reference points for ϕ^{-1} fitting:

$$(s_1 \cdots s_N) = \begin{pmatrix} \tilde{f}_1 & \tilde{f}_2 & \cdots & \tilde{f}_N \\ \Pi_1 & \Pi_2 & \cdots & \Pi_N \end{pmatrix} \quad (15)$$

where N - number of elements in training set. For future discussion, we will denote each unique Π in training and testing sets as reference spectrum. Note, that each values of parameters for all Π from training set was normalized in accordance to $[E_L; E_R] \rightarrow [0; 1]$. Of course, outputs of network were scaled back during testing.

3.2 Conventional methods

Consider methods, which is traditionally used for measurement results evaluation. The data which were obtained by these methods is a base level to determine NN effectiveness for solving spectrum reconstruction problem. The method of least squares calculates parameters Π as:

$$\tilde{\Pi}_{MLS} = \arg \min_{\Pi} \left\| A\tilde{y}_{\Pi} - \tilde{f} \right\|, \quad (16)$$

where $\|\cdot\|$ - Euclidian norm. Method of Tikhonov regularization expands MLS through additional stabilizer function:

$$\tilde{\Pi}_{MTR} = \arg \min_{\Pi} \left(\left\| A\tilde{y}_{\Pi} - \tilde{f} \right\| + \alpha \|\tilde{y}_{\Pi}\| \right), \quad (17)$$

where $\alpha > 0$ - regularization parameter. It should be remind that using of mathematical model of measurement process gives true values of electrons spectrum. So α can be calculated from [7]:

$$\alpha^* = \arg \min_{\alpha} \left(\frac{\|y - \tilde{y}_{\alpha}\|}{\|y\|} \right). \quad (18)$$

In the work we applied Nelder-Mid simplex method numerical solution of (16), (17) and (18).

4 Algorithm for evaluation methods preparing and testing

4.1 Comparison approach

Implemented models of spectrum, measurement process and methods for data evaluation compose computational experiment (Fig. 3 shows sequential diagram). The experiment aim is comparison of methods for spectrum reconstruction. The approach which was used to build experiment uses Monte-Carlo technique: system generate measurement results, each methods reconstruct spectra using samples of depth-charge cure, system calculates statistical characteristics of reconstruction error. Computational experiment consists of three steps: preparation, main part (loop Common) and results interpretation.

Preparation of an experiment includes setting parameters of models and methods. Main part is a series of subexperiments with varied measurement uncertainty ε . Each of them contains two steps: training of NN and selected methods comparison. Both processes include generation of pairs $s = (\tilde{f}, \Pi)$ which is based on predefined set of Π . But these sets of reference spectrum are different. Testing procedure (loop Data Evaluation) repeats sampling of \tilde{f} , evaluates appropriate Π by each method and collects reconstruction error based on truth and calculated spectra based on proposed set of indicators. The results processing step aims to build relationships that show correlations between accuracy of spectrum reconstruction and varied error of measurement.

Software for experiment execution implemented in MATLAB with Neural Network Toolbox (function `newgrnn` as NN), Optimization Toolbox (function `fminsearch` as MLS and MTR). In order to speed up computational experiment, software was executed on high performance cluster [16] with Distributed Computing Toolbox.

4.2 Comparison indicators

In order to assess the effectiveness of methods for reconstruction of beam energy characteristics we suggested set of indicators. The set consists of the standard statistical estimates of data evaluation error and indicator of methods reliability. There are two indicators type: mismatch along energy axis (estimate shift of reconstructed spectrum along horizontal axis) and common indicator. Consider details of each indicators.

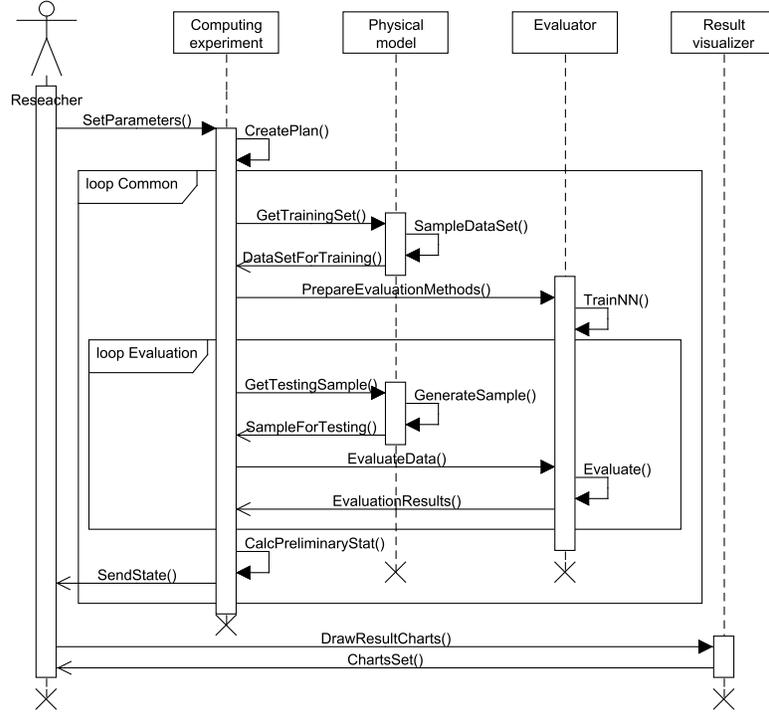


Fig. 3. Sequential diagram of computer experiment

1. *Mismatch along energy axis.* Average $M(r)$ and standard deviation σ_r of distance along intensity axis between reconstructed and true spectra are based on:

$$r = \frac{1}{n} (y - \tilde{y})^2; \quad (19)$$

2. *Common characteristics.* Probability of method failure P . We suppose that the method failure is a case when applying mathematical methods leads to impossible (due to physical laws) solution, i.e. the solution breaks condition (12). It is obvious that value $1 - P$ characterize method reliability.

5 Results and discussions

5.1 Parameters of computation experiment

In order to evaluate methods effectiveness with suggested indicators we made computational experiment with parameters shown in Table 1.

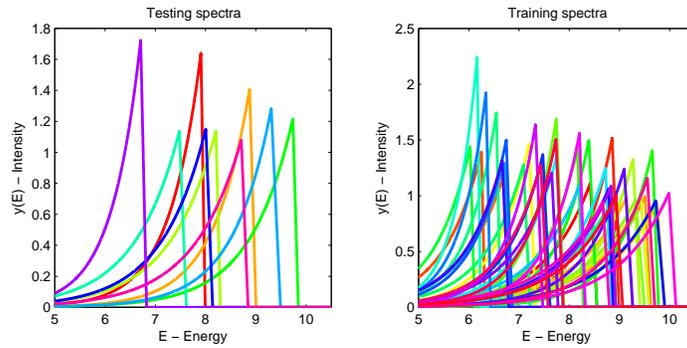
The training and testing sets include reference spectra with parameters:

Table 1. Common experiment parameters

Parameter	Value
Characteristic for measurement	depth-charge curve
Absorbers material	Aluminum ($Z = 13$, $Am = 27$)
Absorber's width (Δx)	$0.4 g/cm^2$
Device total width (x_R)	$6 g/cm^2$
Uncertainty (ε)	Varied from 0% to 30%, step 1%
$[E_L; E_R]$	$[0MeV; 10.2MeV]$
E_w of reference spectra	Randomly from 2% to 10% of E_p
$y(E)$ discretization step (ΔE)	$0.05MeV$
Number of reference spectra	9000 (training) and 41000 (testing)

$$\begin{aligned}
E_p &= r_1, & r_1 &\sim U[E_L, E_R], \\
E_{max} &= E_p(1 + 2r_2), & r_2 &\sim U[0.01, 0.02], \\
E_s &= E_p - \frac{\ln 0.1}{\ln 0.5} r_3, & r_3 &\sim U[0.01, 0.08].
\end{aligned} \tag{20}$$

Fig. 4 shows examples of sampled reference spectra. Number of the spectra for training and testing sets is reduced, but proportion saved. As shown on Fig. 4 and in Table 1 the testing set is bigger than training set. It is necessary to get appropriate assessment of method based on NN with influence of retraining.

**Fig. 4.** Reference spectra for a) NN training and b) methods testing

As shown in Table 1 the device consists of 15 absorbers. This configuration is chosen based on previous research [18] which was aimed to find optimal discretization step of depth-charge curve for spectrum reconstruction by NN. It should be mention that in works [17, 18] sets for methods testing and preparation based on reference spectra with fixed E_w parameter and same maximum $h = 1$. Therefore, seeking of optimal absorbers width is open for future research.

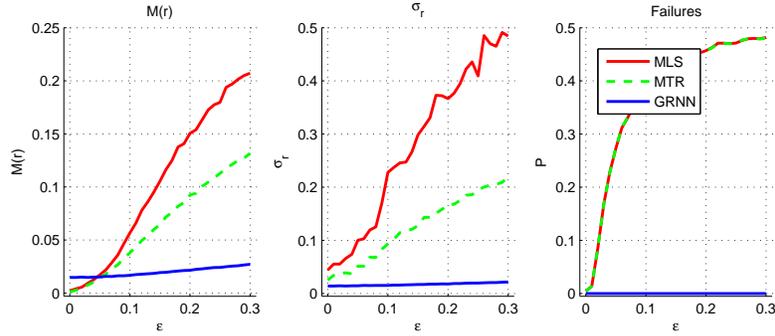


Fig. 5. Results of methods comparison

5.2 Results and discussion

Fig. 5 contains obtained dependencies, which describe relation between methods evaluation error and measurement uncertainty. The charts 5a and 5b based on indicator (19). Chart 5c shows probability of method failure.

For MLS and MTR experiment proves expected results. Methods are sensitive to uncertainty in input data. Fig. 5a and 5b show that error of MLS and MTR solutions rapidly grows with increasing of ϵ . With respect to probability of failure, both methods demonstrated almost equal inefficiency. It can be mean that stabilizing additions in MTR does not affect to the method reliability. It should be mention that the reason of MLS and MTR error for $\epsilon = 0\%$ is discretization inaccuracy which appears when transforming integral (1) to system (4).

As an opposite to conventional methods, the solutions obtained by NN have smaller dependency between evaluation error and input data uncertainty. Furthermore as shown on Fig. 5a, 5b the GRNN evaluates spectra more accurate than MLS and MTR for measurement uncertainty more than 5-7%. The main advantage of NN method is that GRNN reconstruct beam spectrum parameters with no failures (see Fig 5c), i.e. all obtained solutions are compliance with physical laws.

6 Conclusion

The work shows GRNN method effectiveness for solving inverse dosimetry problem of electron spectrum reconstruction using depth-charge curve. The main advantages of proposed technique compared to conventional methods is allowance to apply additional solutions conditions. It lids to getting robust evaluation method. As shown in the work methods based on NN can be used for building on-line energy monitoring systems in centers of radiation technologies.

Furthermore, we proposed comparison approach based on Monte-Carlo technique and set of effectiveness indicators. The approach allows testing different

types of evaluation methods and can be used for methods optimization in order to select or apply technique for industrial problems solving.

References

1. Standard ISO/ASTM 51649-2005(E). Practice for dosimetry in an electron beam facility for radiation processing at energies between 300 keV and 25 MeV. United States, 30 p. (2005)
2. ICRU Report 35. Electron beams with energies between 1 and 50 MeV. United States, 160 p. (1984)
3. Fuochi P.G., Lavallo M., Martelli A., Corda U., Kovacs A., Hargittai P., Mehta K., Electron energy device for process control, Radiation Physics and Chemistry, Volume 67, pp. 593-598 (2003)
4. Fuochi P.G., Lavallo M., Martelli A., Corda U., Kovacs A., Hargittai P., Mehta K., Energy device for monitoring 4-10 MeV industrial electron accelerators, Nuclear Instruments and Methods in Physics Research A, Volume 546, pp. 385-390 (2005)
5. M. Lavallo, P.G. Fuochi, A. Martelli, U. Corda, A. Kovacs, K. Mehta, and F. Kuntz, Energy Monitoring Device for Electron Beam Facilities, International Topical Meeting on Nuclear Research Applications and Utilization of Accelerators, Conference proceedings, Vienna (2009)
6. Vanzha S.A., Nikiforov V.I., Pomatsalyuk R.I., Tenishev A.Eh., Uvarov V.L., Shevchenko V.A., Shlyakhov I.N., Development "radiation shadow" technique for regime monitoring of product sterilization by electron beam, Problems of Atomic Science & Technology. Series "Nuclear Physics Investigations", Volume 2(53), pp. 150-153 (2010)
7. Petrov Yu.P., Sizikov V. S., Well-Posed, Ill-Posed, and Intermediate Problems with Applications, V.S.P. Intl Science, Leiden, Netherlands, 234 p. (2005)
8. Haykin S, Neural networks: a comprehensive foundation, 2nd edn. Prentice Hall, Englewood Cliffs, United States, 936 p. (1999)
9. Michael M. Li, Brijesh Verma, Xiaolong Fan, Kevin Tickle, RBF neural networks for solving the inverse problem of backscattering spectra, Neural Computing and Applications, Volume 17, pp. 391-397 (2008)
10. Michael M. Li, William Guo, Brijesh Verma, Kevin Tickle, John OConnor, Intelligent methods for solving inverse problems of backscattering spectra with noise: a comparison between neural networks and simulated annealing, Neural Computing and Applications, Volume 18, pp. 423-430 (2009)
11. Barradas N. P., Vieira A., Artificial neural network algorithm for analysis of Rutherford backscattering data, Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics 62, pp. 58185829 (2000)
12. Barradas N.P., Patricio R.N., Pinho H.F.R., Vieira A., A general artificial neural network for analysis of RBS data of any element with Z between 18 and 83 implanted into any lighter one- or two-element target, Nuclear Instruments and Methods in Physics Research B, Volumes 219-220, pp. 105-109 (2004)
13. Adadurov A., Lazurik V., Rogov Yu., Tokarevskii V., Shechenko S., Spectrometry of Intense Fluxes of Gamma Radiation by Means of the Method of Capsule-Absorbers, IEEE Nuclear Science Symposium and Medical Imaging Conference, Conference Publications, Anaheim, Unated States, pp. 17 (1996)
14. Lazurik V.T., Lazurik V.M., Popov G., Rogov Yu., Zimek Z., Information System and Software for Quality Control of Radiation Processing. IAEA: Collaborating

- Center for Radiation Processing and Industrial Dosimetry, Warsaw, Poland, 220 p. (2011)
15. Specht Donald F., A General regression neural network, IEEE Transactions on Neural Networks, Volume 2(6), pp. 568–576 (1991)
 16. Baiev O., Didenko I., Lazurik V., Mishchenko V., Towards the questions on planing the development of the department compute cluster, Proceedings of ICTERI-2011, Kherson, Ukraine, pp. 27–28 (2011)
 17. Baiev O., Lazurik V., Advantages of neural networks for deriving an electrons spectrum from depth-charge curve, “IEEE Nuclear Science Symposium and Medical Imaging Conference”, Conference Publications, Valencia, Spain, pp. 1395-1397 (2011)
 18. Baiev O.U., Lazurik V.T., Discretization grid of depth-charge curve selecting for electrons beam spectrum reconstruction problem, Bulletin Kherson National Technical University, Value 3(42), pp. 62–66 (2011)

Lazy Parallel Synchronous Composition of Infinite Transition Systems

Yuliia Romenska and Frédéric Mallet

Université Nice Sophia-Antipolis Aoste Team Project (INRIA/I3S),
Sophia Antipolis, France

`Yuliia.Romenska@inria.fr`, `Frederic.Mallet@unice.fr`

Abstract. Embedded System Design is becoming a field of choice for Model-Driven Engineering techniques. On the engineering side, models bring an abstraction of the code that can then be generated (and regenerated) at will. On the semantic side, they bring a reasoning framework to guarantee or verify properties on the generated code. We focus here on the Clock Constraint Specification Language, initially defined as a companion language of the UML Profile for MARTE. More specifically, we define a state-based representation of CCSL operators. To deal with unbounded operators, we propose to use lazy evaluation to represent *intentionally* infinite transition systems. We provide an algorithm to make the synchronized product of such transition systems and we study its complexity. Even though the transition systems are infinite, the result of the composition may become finite, in which case the (semi)algorithm terminates and exhaustive analysis becomes possible.

Keywords. Multiform logical time, synchronized product, lazy evaluation, MARTE CCSL.

Key terms. FormalMethod, VerificationProcess, MathematicalModel, SpecificationProcess.

1 Introduction. Context and Goal of the Project

In the model-driven approach to embedded system engineering, application and architecture models are developed and refined concurrently, and then associated by allocation relationships. The representation of requirements and constraints in this context becomes itself an important issue, as they guide the search for optimal solutions inside the range of possible allocations. One of the important aspects of embedded system modeling is to capture the functional and non-functional requirements as well as the constraints (functional and non-functional) imposed by the execution platform through the allocation.

Multiform logical time is a flexible notion of time suitable for both functional and extra-functional properties that supports an iterative refinement process. Logical time considers time bases that can be generated from sequences of events

not necessarily regular in physical time (as the usual meaning suggest). Some of the essence of multiform logical time was captured and encapsulated into a dedicated language called the Clock Constraint Specification Language (CCSL) [1,2]. CCSL was initially defined as a companion language of the UML profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE) [3].

CCSL has arisen from different models in an attempt to abstract away the data and the algorithm and to focus on events and control. Even though CCSL was initially defined as the time model of the UML profile for MARTE, it has now become a full fledged domain-specific modeling language for capturing chronological, causal and timed relationships and is now developed independently. It combines constructs from the general net theory and from the synchronous languages [4]. It is based on the notion of clocks which is a general name to denote a totally ordered sequence of event occurrences. It defines a set of clock relations and expressions. Some CCSL operators are bounded, others are unbounded. The bounded operators can be represented with finite Boolean transition systems. Unbounded operators require a specific symbolic representation.

Until then, the clock calculus on CCSL specification was performed step by step up to a predefined number of steps. This work is an attempt to support exhaustive analysis of CCSL specification. When CCSL operators are represented as transition systems, their composition is the synchronized product of the transition systems. However, this causes termination problems when the transition systems have an infinite number of states. In this paper, an algorithm for the parallel execution of automata representing CCSL operators is proposed. It has been implemented in a prototype tool. This algorithm supports CCSL unbounded operators. The infinite data structure is unfolded on demand using a lazy evaluation technique. This is a significant evolution on previous verification techniques for CCSL [5,6] that were only considering a subset of operators a priori bounded.

2 Contribution

The main contribution is to propose an encoding based on lazy evaluation to represent CCSL unbounded operators. The second contribution is to propose an algorithm to build the synchronized product of such automata. The (semi)algorithm terminates when the composition of unbounded automata becomes bounded.

In this work, the main operators of the clock constraint language were considered. For each basic expression and each relations of the kernel language, a transition system is proposed. Each transition is labeled by the set of clocks that must tick for the transition to be taken. Clocks can stall, tick or be dead. When a clock is dead, it cannot tick anymore. A path in the automaton is then an infinite word on the alphabet of powersets of clock names.

The automata representing the unbounded CCSL operators consist of an infinite number of states and therefore transitions (even though each state has a finite number of outgoing transitions). For those operators, the lazy evaluation technique was applied. It allows postponing the construction of the state of an unbounded automaton to the moment when it is actually needed. In very

frequent cases, the specification becomes bounded and the intentional infinite representation is never actually expanded.

On these transition systems, we apply the classical synchronized product of transition systems [7]. In the worst case (when automata are independent) the composition is very costly, exponential in the number of clocks. In some (frequent) cases, the cost of composition is much better, even though we have not identified yet the exact cases where the composition is tractable.

3 Related work and inspirations

3.1 Timed automata

The formalism of timed automata [8] has been designed to allow the specification and verification of real-time systems. It extends the concept of finite ω -automata by establishing time constraints. One of the main notions of the timed automata theory is a clock (not to be confused with the notion of clocks introduced in CCSL). In a timed transition table the selection of the next state depends on an input symbol and the time reading of the symbol. For this purpose each transition table is associated with the set of real-valued clocks. A clock can be set to zero simultaneously with the execution of one of the transition. At any instant the values of such clocks are equal to the time elapsed since the last time they were reset. The transition can be executed *only if* the value of the clock satisfies the constraint associated with this transition. Therefore in the theory of the timed automata a clock is an entity intended for determination of time which elapsed since the last execution of the transition and the setting the value of the clock to zero.

To answer the question if this formalism can be applied for representation of the CCSL language basic constructions, the definition of a clock in CCSL time model must be considered. In terms of CCSL time model a clock is a set of ordered instants. Each clock has a lifetime limited by birth and death instants. Formally, a *Clock* c is a tuple $(\mathcal{I}_c, \prec_c, c^\uparrow, c^\downarrow, \equiv_{c\downarrow})$ where \mathcal{I}_c is a sequence of instants (it can be infinite), c^\uparrow, c^\downarrow are birth and death instants respectively such that $\mathcal{I}_c \cap \{c^\uparrow, c^\downarrow\} = \emptyset$, $\equiv_{c\downarrow}$ is a coincidence relation and \prec_c is an order relation on $\mathcal{I}_c \cup \{c^\uparrow, c^\downarrow\}$. All instants of clocks are strictly ordered, the birth instant precedes all the other instants of the clock and every instant precedes the death. If the set of instants \mathcal{I}_c is infinite then the death instant is not necessary. \mathcal{I}_c represents the occurrences or ticks of the clock c .

Thus we can see that the notions of clock in the terms of timed automata formal model and the clock constraint specification language are radically different. Timed Automata, clocks captured physical real-valued time properties, whose value is within a dense time interval. All time clocks evolve at the same rate (without drift). In CCSL, clocks represent logical time properties. The unbounded nature precisely comes from the relative (unbounded) drifts between the clocks that evolve at their own independent rhythm.

3.2 Synchronous Data Flow, Marked Graphs

Synchronous Data Flow (SDF) [9, 10] is a special case of the dataflow model of computation. Its main characteristic is that the flow of control is completely predictable at compile time. The main components of SDF are the actors, tokens, and arcs. Production and consumption of tokens by actors allow modeling of relative rates of events. In synchronous dataflow numbers of consumed and produced tokens are constant throughout the execution. To avoid the overflow of resources and to maintain a balanced system, the scheduler must fire the source and destination components at different rates. Such systems can then be used to capture the relative drift between CCSL clocks.

Safety analysis on SDF graphs is a way to determine whether the system remains bounded or not. Such techniques could be used to study boundness issues for CCSL specifications. However, this is not the concern of this paper. We assume that the composition is bounded and propose an algorithm to build the synchronized product.

3.3 Synchronized Product of Transition Systems

When CCSL operators are expressed as transition systems, their parallel composition simply is the synchronized product of the transition systems [7, 11, 12]. Synchronization vectors are used to decide which transition systems must synchronize on which transitions. Synchronization vectors allows the specification of purely asynchronous compositions (where only one single system is fired at each step) to purely synchronous compositions (where all the automata must fire one transition at each step), and all the intermediate synchronization schemes. The main difference here is that the number of states may be infinite and we use lazy evaluation to dynamically expand the states whenever they are required to build a new composite state. The composition algorithm terminates only when the synchronized product becomes finite. In [13], there was an initial attempt to build the synchronized product of unbounded CCSL operators. In that work, the automata were folded using *extended automata* (with unbounded integer variables) rather than lazy evaluation. Therefore, the algorithm to compute the synchronized product was always guaranteed to terminate. However, deciding whether the result was finite or not would then require using integer linear programming techniques.

4 The Clock Constraint Specification Language

This section briefly introduces the logical time model of the *Clock Constraint Specification Language* (CCSL). A technical report [1] describes the syntax and the semantics of a kernel set of CCSL constraints.

A clock c is a totally ordered set of instants, I_c . In the following, i and j are instants. A time structure is a set of clocks C and a set of relations on instants $I = \bigcup_{c \in C} I_c$. CCSL considers two kinds of relations: causal and temporal

ones. The basic causal relation is causality/dependency, a binary relation on $I : \preceq \subset I \times I$. $i \preceq j$ means i causes j or j depends on i . \preceq is a pre-order on I , i.e., it is reflexive and transitive. The basic temporal relations are precedence (\prec), coincidence (\equiv), and exclusion ($\#$), three binary relations on I . For any pair of instants $(i, j) \in I \times I$ in a time structure, $i \prec j$ means that the only acceptable execution traces are those where i occurs strictly before j (i precedes j). \prec is transitive and asymmetric (reflexive and asymmetric). $i \equiv j$ imposes instants i and j to be coincident, i.e., they must occur at the same execution step, both of them or none of them. \equiv is an equivalence relation, i.e., it is reflexive, symmetric and transitive. $i \# j$ forbids the coincidence of the two instants, i.e., they cannot occur at the same execution step. $\#$ is irreflexive and symmetric. A consistency rule is enforced between causal and temporal relations. $i \preceq j$ can be refined either as $i\pi j$ or $i \equiv j$, but j can never precede i . We consider here discrete sets of instants only, so that the instants of a clock can be indexed by natural numbers. For a clock $c \in C$, and for any $k \in N_{>0}$, $c[k]$ denotes the k^{th} instant of c .

Specifying a full time structure using only instant relations is not realistic since clocks are usually infinite sets of instants. Thus, an enumerative specification of instant relations is forbidden. The Clock Constraint Specification Language (CCSL) defines a set of time patterns between clocks that apply to infinitely many instant relations.

4.1 The kernel relations

Table 1 gives a full list of the basic clock relations provided in the CCSL kernel. For each of them the automaton was built. It is supposed that the automaton can fire only if one of the participant clocks in CCSL operator ticks.

Table 1. Basic relations defined in the CCSL kernel (a and b are clocks, not instants).

Ref	Name	Kind of relation	Notation
R1	Subclocking	Synchronous	$a \boxed{\subset} b$
R2	Coincidence	Synchronous	$a \boxed{=} b$
R3	Precedence	Asynchronous, unbounded	$a \boxed{\prec} b$
R4	Strict Precedence	Asynchronous, unbounded	$a \boxed{\prec} b$
R5	Exclusion	Asynchronous	$a \boxed{\#} b$

Coincidence According to the considered relation the clocks a and b always tick simultaneously ($a \boxed{=} b$), it is defined as $(\forall k \in \mathbb{N}^*)(a[k] \equiv b[k])$.

Subclocking $a \boxed{\subset} b$ defines a as being a subclock of its superclock b . Every instant of the subclock occurs synchronously with one of the instants of the superclock: $(\forall k \in \mathbb{N}^*)(\exists i \in \mathbb{N}^*)(a[k] \equiv b[i])$.

Exclusion $a \boxed{\#} b$. The clocks connected with this relation cannot have coincidence instants: $(\forall k \in \mathbb{N}^*)(\forall i \in \mathbb{N}^*)(a[k] \# b[i])$.

Precedence $a \boxed{\prec} b$ is the index-dependent relation. This operator is unbounded. Every instant of clock a has to precede the instant of clock b with the same index $(\forall k \in \mathbb{N}^*)(a[k] \prec b[k])$.

Strict Precedence $a \boxed{\preceq} b$. This relation is a severer version of the previous one in the sense that the instants of the clocks a, b with the same indices cannot be equal: $(\forall k \in \mathbb{N}^*)(a[k] \preceq b[k])$.

4.2 The kernel expressions

A CCSL specification consists of clock declarations and conjunctions of clock relations between clock expressions. A clock expression defines a set of new clocks from existing ones. Most expressions deterministically define one single clock. Table 2 gives a list of the CCSL kernel expressions.

Table 2. Basic expressions defined in the CCSL kernel.

Ref	Name	Kind of expression	Notation	Textual form
E1	Inf	Mixed, unbounded	$a \wedge b$	$a \text{ glb } b$
E2	Sup	Mixed, unbounded	$a \vee b$	$a \text{ lub } b$
E3	Defer	Mixed	$a (ns) \rightsquigarrow b$	$a \text{ deferred } b \text{ for } ns$
E4	Sampling	Mixed	$a \mapsto b$	$a \text{ sampling } b$
E5	Strict sampling	Mixed	$a \rightarrow b$	$a \text{ strictlySampled } b$
E6	Intersection	Synchronous	$a * b$	$a \text{ clockInter } b$
E7	Union	Synchronous	$a + b$	$a \text{ clockUnion } b$
E8	Concatenation	Synchronous	$a \bullet b$	$a \text{ followedBy } b$
E9	Waiting	Synchronous	$a \lambda_n b$	$a \text{ wait } n \text{ } b$
E10	Preemption (UpTo)	Synchronous	$a \dagger b$	$a \text{ upto } b$

Sampling $a \mapsto b$. The sampling expression ticks in coincidence with the tick of the base clock immediately following a tick of the trigger clock and after it dies. In the considered case, the trigger clock is b and the base clock is a . The textual syntax of this expression is represented as $c = a \text{ sampling } b$. In Figure 1 the automaton is given, where input symbol c is equal to the result clock of the expression. The notation $\{a, b, c\}$ denotes that the automaton remains in state s_2 if a , b and c tick all together simultaneously. if b and c tick simultaneously without a then, the automaton goes back to state s_1 . If a ticks alone, it stays in s_2 . All other cases are forbidden by the semantics of the operator.

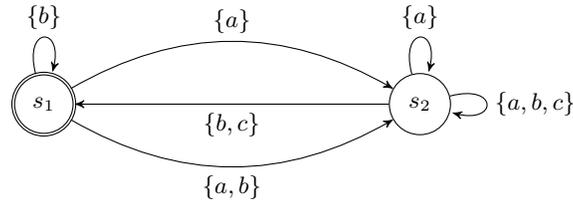


Fig. 1. The automaton for sampling expression

Strict Sampling $a \rightarrow b$. The expression is a strict version of the previous one where c is emitted when the automaton is in state s_1 , and a and b tick simultaneously.

Waiting $a \lambda_n b$. The resulting clock ticks only once after a special number given as a parameter of the base clock, and then the resulting clock dies. $c = a \text{ wait } n b$, where n is a given parameter (it is a natural number).

Preemption (UpTo) $a \downarrow b$. The resulting clock ticks in coincidence with a , it dies as soon as b starts to tick: $c = a \text{ upto } b$.

Union This expression is non-terminating and index-independent. Its result is a clock with set of instants which is a union of the instants sets of the clocks-parameters that participate in the expression: $c = a + b$.

Intersection The result of this index-independent expression is the clock which ticks each time when the clocks-parameters tick simultaneously: $c = a * b$.

Concatenation $a \bullet b$. The expression is terminating. The resulting clock ticks in coincidence with the first clock-parameter a . After death of a it starts to tick in coincidence with the second clock-parameter b . It should be noted that this expression is valid only if the first clock eventually dies, i.e. $a \downarrow$ is specified.

Defer (Delay) $a (ns) \rightsquigarrow b$. The parameter of the expression are a (the base clock), b (the delay clock) and ns that represents a sequence of elements from $\mathbb{N}_{>0}$. The sequence of the natural numbers can have an infinite part. Let $ns[i]$ be the i^{th} element of the sequence. We assume that if $i \geq 0$ then ns has an infinite part, if $0 \leq i < p$ there are p elements in the sequence. Every tick of the base clock starts up the counter for respective element of the sequence. For every tick of the delay clock the relative counter is decreased. When the counter reaches 1 the respective instant of clock b occurs. The textual form of the expression is $c = a \text{ deferred } b \text{ for } ns$.

Sup $a \vee b$. The expression is index-dependent. The expression $a \vee b$ defines a clock that is slower than both a and b and whose k^{th} tick is coincident with the later of the k^{th} ticks of a and b . The formal definition is presented as: $(a \boxtimes (a \vee b))(b \boxtimes (a \vee b))(\forall c \in C) : (a \boxtimes c) \& (b \boxtimes c) \Rightarrow ((a \vee b) \boxtimes c)$. This is a typical example of unbounded transition system (with an infinite number of states).

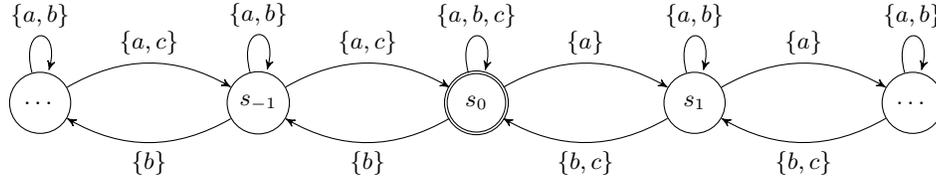


Fig. 2. The automaton for sup expression

Inf $a \wedge b$. This is index-dependent and unbounded. It is the dual of the previous one. The result of the expression is the slowest of faster clocks. It means that the defined clock is faster than both a and b , the k^{th} tick of this clock occurs in coincidence with the earlier of the k^{th} ticks of both a and b . $((a \wedge b) \boxtimes a)((a \wedge b) \boxtimes b)(\forall c \in C) : (c \boxtimes a) \& (c \boxtimes b) \Rightarrow (c \boxtimes (a \wedge b))$ is the formal definition.

4.3 Unbounded CCSL operators

Lazy evaluation or call-by-needed is an evaluation strategy that delays the evaluation of an expression until its value is actually needed. This approach allows construction of potentially infinite data structures and avoids repeated evaluations. To construct the algorithm of the parallel execution of several automata, it is necessary to have the possibility to work with infinite data structures (transition systems with an infinite number of states). Lazy evaluation provides the apparatus for this task. The CCSL has four basic unbounded operators which can be

represented as infinite automata: the precedence relation, the strict precedence relation, the inf expression (the fastest of slower clocks) and the sup expression (the slowest of faster clocks).

Example 1. Let us consider as an example the automaton for the strict precedence relation (Fig. 3).

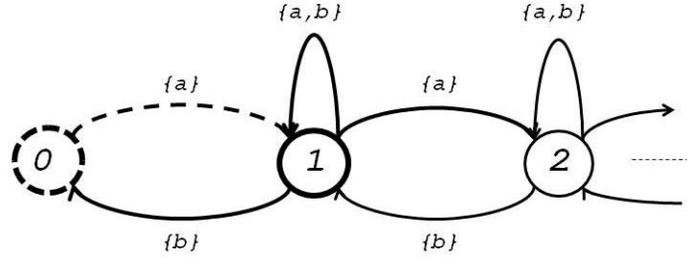


Fig. 3. The automaton for strict precedence relation

If we only consider the possible outgoing transitions of each state, we can distinguish the initial state (0 on Fig. 3) from the other ones. In the initial state, only a can tick and must tick alone. In the other states, if a ticks alone, we must progress to the right (increasing an unbounded counter), if b ticks alone, we must go back to the left (decreasing the counter). If a and b tick simultaneously, we remain in the same state. We assume that each state has a default transition to itself, when nothing happens.

5 The synchronized product: an algorithm

We assume that the input for the algorithm is a given finite set of automata; each of them can be either finite or infinite. The output is the automaton that is the composition of the input ones. We denote the resulting automaton through the tuple R . Let us introduce the set St which is the set of all unconsidered states of the resulting automaton. At the beginning of the algorithm execution, the initial states of input automata are considered as the current ones. The first unconsidered state of the resulting automaton is the state formed from current ones. For each considered current state of input automata the set of available transitions from the given state is calculated. For all input symbols of each of the calculated transitions the consistent solution is computed, which is the set such that every original symbol is its subset. If the solution exists, the new state is calculated from the states of input automata, in which we can go on the appropriate input symbols that are the subset of the found solution. If the computed new state does not belong to St , it is added to the set of unconsidered states. The resulting automaton is completed by the new state

and the appropriate transition, which has the input symbol that is equal to the respective solution. If the solution cannot be found, then we take a state from St , in the input automata the appropriate states are considered as current. The process is repeated while the set St is not empty.

5.1 The Formal Definitions

We introduce the formal definitions that are used to describe the algorithm.

C is the (finite) set of clocks and $|C|$ is its cardinality;

$state : C \rightarrow StateDomain = \{STALL, TICK, DEAD\}$ that denotes the state of a clock.

$A = \{A_i\}$: the ordered set of input automata for the composition.

$A_i = (C_i; S_i; I_i; moveOn_i; s_{i0} \in S_i; cur_i \in S_i; availTrans_i)$, where

$C_i \subset C$: the ordered set of clocks of the i^{th} automaton;

S_i : the set of states of the i^{th} automaton;

$s_{i0} \in S_i$: the initial state of the automaton;

$moveOn_i : S_i \times 2^{C_i} \rightarrow S_i$ is the transition function;

$cur_i \in S_i$: the current considered state;

$availTrans_i : S_i \rightarrow 2^{2^{C_i}}$: gives a set of available configurations for a given state.

The resulting composite automaton can be defined as follows:

$R = (C; S_R; moveOn_R; s_{0R} \in S_R; cur_R \in S_R; availTrans_R)$, such that:

C : the set of clocks of the composite automaton is equal to the set of global clocks;

$S_R = \{s_R : s_R = (s_1, \dots, s_{|A|}), s_i \in S_i, \forall i = 1..|A|\}$: each element of the set of states of the resulting automaton consists of the states of input automata; the element of the considered composite state corresponds to the automaton with the same number;

$moveOn_R : S_R \times 2^C \rightarrow S_R$ is the transition function for the resulting automaton;

$s_{0R} \in S_R$: the initial state;

$cur_R \in S_R$: the current considered state;

$availTrans_R : S_R \rightarrow 2^{2^C}$: this function returns the set of input symbols for the available transitions of the composite automaton.

The set of found solutions is represented by the set $SOL = \{sol : sol = (st_1, \dots, st_{|C|}), st_i \in \{STALL, TICK, DEAD\}, \forall i = 1..|C|\}$.

$St = \{s_R : s_R \in S_R\}$ is the set of considered states of the resulting automaton.

$Get : Set \rightarrow element, element \in Set$: the function that returns an element of the given set.

$Register : C \times \{1..|A|\} \rightarrow 2^C$. This function returns an input symbol for the i^{th} automaton based on the states of the global set clocks.

$index : 2^C \times C \rightarrow \mathbb{N}$, the position of a clock in an ordered set of clocks.

5.2 The Composition Algorithm

Global variables.

```

indexTable:=0:    indexTable ∈ ℕ;
array StateDomain temp[]: ∀i ∈ {0, ..., |C|-1}, temp[i] ∈ StateDomain;
array StateDomain cur_sol[]: ∀i ∈ {0, ..., |C|-1}, cur_sol[i] ∈ StateDomain;
array StateDomain OptionsTable[][]: ∀j ∈ {0, ..., |C|-1}, ∀i ∈ {0, ..., 3|C|-1},
    OptionsTable[i][j] ∈ StateDomain;
array Boolean SolutionsTable[][]: ∀j ∈ {0, ..., |A|-1}, ∀i ∈ {0, ..., 3|C|-1},
    SolutionsTable[i][j] ∈ {true, false};
cur_solution ∈ SOL
new_stR ∈ SR

```

The main purpose of the following function is to build the composite automaton from the given set of the input ones. It uses three other functions: *buildSolutions()*, *getSolutions()* and *buildOptionsTable()*.

```

function composition(){
    // the initial state is the cartesian product of all the initial states
1.  s0R := s00 × ... × s0|A|-1; St:={s0R};

2.  while(St ≠ ∅){
3.      curR:={GetElement(N)};
4.      SOL:=buildSolution(curR);

5.      while(SOL ≠ ∅){ //for each solution
6.          cur_solution:=Get(SOL);

7.          // set clocks to the appropriate states
8.          for(k:=0; k<|C|; k:=k+1){ C[k]:=cur_solution[k]; }

9.          //form a composite state from the states of the input automata in which
          // it is possible to go with the respective input sets
          for(i:=0; i<|A|; i:=i+1){ new_sti := moveOn(curi, Register(C,i)); }

10.         // include the new created state if it is not yet included
          if(new_stR ∉ SR){
11.             SR:={SR ∪ new_stR};
12.             IR:={IR ∪ cur_solution};
13.         }

14.         //include the created state in the set of unconsidered states
          if(new_stR ∉ St){ St:=St ∪ new_stR; }

15.         //set the current states to the previous positions
          for(i:=0; i<|A|; i:=i+1){ curi:={curR[i]}; }

16.         SOL:=SOL \ cur_solution;
17.     }
18.     N:={N \ curR};
19. }
}

```

The following function calculates the set of possible solutions. It builds a table of all options (*OptionsTable*) for the states of the clocks. The number of rows of this table is equal to $3^{|C|}$. The base is equal to three because all three possible states are considered (*STALL*, *TICK*, *DEAD*). The number of columns is equal to the number of the clocks in the global set. Also, the table for finding solutions is defined (*SolutionsTable*). The number of rows is equal to $3^{|C|}$, the number of columns corresponds to the number of input automata. If the input set defined in a row of *OptionsTable* is available for the input automaton, then the value of an element of *SolutionsTable* situated in the same row as the input set is set the value true. When the *SolutionsTable* is completed (all available input symbols of input automata have been considered), the function *getSolutions()* is invoked to find the set of solutions SOL using the data of *SolutionsTable*.

```
function buildSolutions( $s_R$ ){
20.  buildOptionTable(0);

    //Initially there are no solutions
21.  for( $r:=0$ ;  $r<3^{|C|}$ ;  $r:=r+1$ ){
22.      for( $i:=0$ ;  $i<|A|$ ;  $i:=i+1$ ){ SolutionsTable[ $r$ ][ $i$ ]:=false; }
23.  }

24.  for( $i:=0$ ;  $i<|A|$ ;  $i:=i+1$ ){
    // receiving the set of input sets for all available transitions
    // of the current state of the appropriate  $i^{th}$ 
    // automaton  $n$  is a number of available transitions
25.       $I_i^n := availTrans_i(s_R[i])$ ;

    //assignment of the value true to an element of SolutionTable
26.      for( $k:=0$ ;  $k<n$ ;  $k:=k+1$ ){ setMark( $I_i^k$ ) }
27.  }

28.  return getSolutions();
}
```

The function based on the values of the completed table of possible solutions (*SolutionsTable*) determines whether there are solutions.

```
function getSolutions(){
29.  SOL:= $\emptyset$ ;

    //for all the rows in SolutionsTable
30.  for( $r:=0$ ;  $r<3^{|C|}$ ;  $r:=r+1$ ){
31.      boolean isSolution:=true;
32.      for( $i:=0$ ;  $i<|A|$ ;  $i:=i+1$ ){

        // if at least one value in the row of table is equal to false
        // (it means that the given input set is not available for one
        // of automata) then the corresponding row of OptionsTable
        // cannot be considered as solution
```

```

33.     if(SolutionsTable[r][i]==false) { isSolution:=false; }
34.     if(isSolution){
35.         for(k:=0; k<|C|; k:=k+1){
36.             cur_sol[k]:=OptionsTable[r][k];
37.         }
38.         SOL:=SOL ∪ cur_sol;
39.     }
40. }
41. return SOL;
}

```

The function of building the table for all possible options (*OptionsTable*).

```

function buildOptionsTable(index){
42. if(index<0){
43.     temp[index]:=STALL;
44.     buildOptionsTable(index+1);
45.     temp[index]:=TICK;
46.     buildOptionsTable(index+1);
47.     temp[index]:=DEAD;
48.     buildOptionsTable(index+1);
49. } else{
50.     for(k:=0; k<|C|; k++){
51.         OptionsTable[indexOfTable][k]:=temp[k];
52.     }
53.     indexOfTable:=indexOfTable+1;
54. }
}

```

The following function assigns the value *true* to an element of *SolutionsTable*. The element corresponds to the current considered input set defined in the table *OptionsTable*

```

function setMark( $I_i^k$ ){
55. for(r:=0; r<3|C|; r:=r+1){
56.     boolean isDifferent:=false;
57.     for(p:=0; p<| $I_i^k$ |; p:=p+1){
58.         if(OptionsTable[r][index( $I_i^k$ [p])]≠  $I_i^k$ [p]){
59.             isDifferent:=true;
60.         }
61.     }
62.     if(!isDifferent){
63.         SolutionsTable[r][i]:=true;
64.     }
65. }
}

```

5.3 The Time Complexity of the Algorithm

To determine the complexity of the composition algorithm, it is necessary to determine the time complexity of each of the used functions. Let us consider

the function *buildOptionsTable()* (lines 42-54). This function is recursive. The recursion depth is determined by the number of clocks in the global set. Thus, the time complexity is equal to $3^{|C|}$.

The function *getSolutions()* has two nested loops. The outer loop (lines 30-40) goes over all rows of the solution table. The number of iterations is equal to $3^{|C|}$. The nested loop (lines 32-39) is designed to examine the marked input sets placed in the same row of the corresponding table. Since the number of elements in the row is equal to the number of automata, the number of iterations is $|A|$. The complexity of the function given by the product is defined as $3^{|C|} \times |A|$.

The *setMark()* function is represented by two loops. The number of iterations of the outer (lines 55-65) is equal to $3^{|C|}$. The number of iterations of the inner loop (lines 57-60) is determined by the number of elements in the input set of the considered automaton. Because it is equal to the number of clocks involved in the execution of the relevant automaton, the number of repetitions is equal to $|C_i|$.

We define the complexity of the function for building the set of solutions *buildSolutions()*. The initialization phase (lines 21-23) is represented by two nested loops, the time complexity of its implementation is equal to $3^{|C|} \times |A|$. The process of the solutions table completion is presented by lines 24-27. The number of iterations of the outer (line 24-27) is equal to the number of input automata $|A|$. The number of the repetitions of the first inner loop operations depends on the number of elements which belong to the previously found set of input sets of the available transitions for the considered automaton. The size of the set is marked as n . The total time complexity of the function taking into account the previous results is equal to

$$D := 3^{|C|} \times |A| + 3^{|C|} \times |A| + 3^{|C|} + |A| \times n \times 3^{|C|} \times |C_i| \quad (1)$$

The complexity of the developed algorithm can be computed on the basis of our results. The initialization of the initial state of the resulting automaton requires $|A|$ operations. The cycle through all the unconsidered (lines 3-19) composite states is determined by the number of elements in N , the number of iterations is equal to $|St|$. In this loop the considered earlier function for building solutions (line 5) is involved. Its complexity is defined as (1). To examine all built solutions, it is necessary to perform a number of iterations which is equal to the cardinal number of the set $|SOL|$. To assign to all clocks of the global set the corresponding states, it must be $|C|$ operations (lines 8). To form a new composite state (line 9), it is performed $|A|$ repetitions of the loop code. To reset the states of input automata, it is necessary to perform the number of iterations which is equal to the number of input automata. The formula expressing the time complexity of the algorithm is as follows:

$$|A| + |St|(D + |SOL|(|C| + 2 \times |A|)) \quad (2)$$

Since we are considering the worst case in which all the expressions of the clock constraint specification language do not have common clocks, the value n which is equal to the number of all available transitions can be approximated

by the quantity $3^{|C_i|}$, since the number of the available transitions is not greater than the number of all possible combinations of the states of clocks for the considered input automaton. Thus, after the change of the corresponding quantity the representation formula for the time complexity becomes:

$$|A|(1 + |St|(3^{|C|}(2 + \frac{1}{|A|} + 3^{|C_i|} \times |C_i|) + |SOL|(\frac{|C|}{|A|} + 2))) \quad (3)$$

From the obtained result (3) we can conclude that the developed composition algorithm has exponential complexity in the number of clocks involved in the composition.

Additionally, it depends on the number of states in the resulting automaton. When the composition is infinite then the algorithm does not terminate. When the composition is finite, the complexity is exponential in the number of clocks in the worst case.

5.4 The Implementation of the Algorithm

For a software implementation of the algorithm, it is necessary to represent each kind of the built automata in terms of classes and to define the suitable presentation of the correspondent states. There are two types of input automata: finite and infinite. For the latter ones, lazy evaluation is used to compute the next state on demand.

All classes are divided into 5 main, architecturally significant packages: *entities*, *abstractions*, *states*, *automata* and *executors*. The *entities package* contains the primary classes-entities, which are necessary for the implementation of the composition algorithm such as the class for the representation of the clock notion, the class for the resulting automaton. The *abstractions package* stores two main interfaces that specify responsibilities for all automata and states. The *states package* includes all classes intended to work with states of all kinds of automata. The implementation for all previously formally defined automata for the basic CCSL operators is contained in the *automata package*. The *executors package* includes classes-executors that are responsible for organizing the correct interactions of all classes so as to build the synchronized product of the input automata.

6 Conclusion

This work has proposed a data structure based on lazy evaluation to encode the semantics of CCSL operators as transition systems. Then the composition of CCSL operators is computed as the synchronized product of those transition systems. The underlying definitions were implemented in Java. The automaton form representation for basic expressions and relations of the CCSL kernel were proposed. The lazy evaluation was considered and applied for building the unbounded automata of the corresponding operators.

The complexity analysis of the algorithm was conducted. It allowed estimating the time resources needed by the composition algorithm which solves a given computational problem. It was shown that the computational complexity of the developed algorithm is exponential in the number of clocks and is linear with the size of the resulting automaton. Such an automaton can be infinite in which case the (semi)algorithm does not terminate.

However, there are many classical examples where the resulting automaton is finite, and where the actual complexity is much better than the theoretical worst-case computed here. As future work, we intend to qualify useful cases that are tractable in CCSL and therefore would allow model-checking CCSL specifications.

References

1. André, C.: Syntax and Semantics of the Clock Constraint Specification Language (CCSL). Research Report RR-6925, INRIA (2009)
2. Mallet, F.: CCSL: specifying clock constraints with UML/MARTE. *Innovations in Systems and Software Engineering* **4**(3) (2008) 309–314 The original publication is available at www.springerlink.com.
3. OMG: UML Profile for MARTE, v1.0. Object Management Group. (November 2009) formal/2009-11-02.
4. Benveniste, A., Caspi, P., Edwards, S.A., Halbwachs, N., Le Guernic, P., de Simone, R.: The synchronous languages 12 years later. *Proceedings of the IEEE* **91**(1) (January 2003) 64–83
5. Gascon, R., Mallet, F., DeAntoni, J.: Logical time and temporal logics: Comparing uml marte/ccsl and psl. In: *TIME*. (2011) 141–148
6. Yin, L., Mallet, F., Liu, J.: Verification of marte/ccsl time requirements in promela/spin. In: *ICECCS*. (2011) 65–74
7. Arnold, A.: Synchronized products of transition systems and their analysis. In: *ICATPN*. (1998) 26–27
8. Alur, R., Dill, D.L.: A theory of timed automata. *Theoretical Computer Science* **126** (1994) 183–235
9. Lee, E.A., Messerschmitt, D.G.: Static scheduling of synchronous data flow programs for digital signal processing. *IEEE Trans. Computers* **36**(1) (1987) 24–35
10. Lee, E.A., Parks, T.M.: *Readings in hardware/software co-design*. Kluwer Academic Publishers, Norwell, MA, USA (2002) 59–85
11. Arnold, A.: *Finite transition systems - semantics of communicating systems*. Prentice Hall international series in computer science. Prentice Hall (1994)
12. Arnold, A.: Nivat’s processes and their synchronization. *Theor. Comput. Sci.* **281**(1-2) (2002) 31–36
13. Mallet, F.: Automatic generation of observers from marte/ccsl. In: *RSP*. (2012) 86–92
14. Julien DeAntoni, Charles André, R.G.: CCSL denotation semantics. Research Report RR-8000, INRIA (2010)
15. Ptolemy: Synchronous dataflow (February 2011)
16. Wikipedia: Lazy evaluation (December 2012)

Selecting Mathematical Software for Dependability Assessment of Computer Systems Described by Stiff Markov Chains

Vyacheslav Kharchenko^{1,2}, Oleg Odarushchenko², Valentina Odarushchenko¹
and Peter Popov³

¹National Aerospace University “KhAI”, Kharkov, Ukraine

{v.kharchenko, v.odarushchenko}@khai.edu

²RPC “Radiy”, Kirovograd, Ukraine

skifs2005@mail.ru

³Centre for Software Reliability, City University London, United Kingdom

ptp@csr.city.ac.uk

Abstract. Markov and semi-Markov models are widely used in dependability assessment of complex computer-based systems. Model stiffness poses a serious problem both in terms of computational difficulties and in terms of accuracy of the assessment. Selecting an appropriate method and software package for solving stiff Markov models proved to be a non-trivial task. In this paper we provide an empirical comparison of two approaches to dealing with stiffness – stiffness avoidance and stiffness-tolerance. The study includes several well known techniques and software tools used for solving Kolmogorov’s differential equations derived from complex stiff Markov models. In the comparison we used realistic cases studies developed by others in the past: i) a computer system with hardware redundancy and diverse software, and ii) a queuing system with a server break-down and repair. The results indicate that the accuracy of the known methods is significantly affected by the stiffness of the Markov models, which led us to developing a procedure (an algorithm) for selecting the optimal method and tool for solving a given stiff Markov model. The algorithm is, also included in the paper.

Keywords. Markov chains, stiffness, stiffness-avoidance, stiffness-tolerance, computer based systems, availability, multi-fragmentation

Key terms. MathematicalModeling, Method, SoftwareSystems

1 Introduction

Dependability of computer systems is assessed using probabilistic models in which reliability and availability are typically used as measures of interest. Markov chains (MC) [1, 2] are often preferred to reliability block diagrams and fault trees as MC can handle well complex situations such as failure/repair dependencies and shared repair resources [3].

Dependability assessment of complex computer systems is an essential part of the development process as it either allows for demonstrating that relevant regulations have been met (e.g. as in safety critical applications) and/or for making informed decisions about the risks due to automation (e.g. in applications when poor dependability may lead to huge financial losses). Achieving these goals, however, requires accurate assessment. Assessment errors may lead to wrong or suboptimal decisions.

System modellers are often interested in *transient measures*, which provide more useful information than steady-state measures. The main computational difficulty when MC are used is the size of the models (i.e. their largeness), which is known to affect the accuracy of the transient numerical analysis.

It is not unusual for modern complex systems to have a very large state space: often it may consist of tens of thousands of states. Additional difficulty in solving such models is the model *stiffness* [5], which is the focus of this paper. In practice stiffness in models of computer systems is caused by: i) in case of repairable systems the rates of failure and repair differ by several orders of magnitude [4]; ii) fault-tolerant computer systems (CS) use redundancy. The rates of simultaneous failure of redundant components are typically significantly lower than the rates of the individual components [4]; iii) in models of reliability of modular software the modules' failure rates are significantly lower than the rates of passing the control from a module to a module [4].

In practice it is useful to detect the model stiffness as early as possible. If the model is stiff, using a small integration step is usually a necessary step for obtaining an accurate solution. On the other hand, models with moderate stiffness may allow for obtaining accurate solutions without using a small integration step, thus saving computational resources. With some numerical methods decreasing the step of integration may be even counterproductive as it may simply not improve the solution accuracy.

The assessment methods and tools must provide high confidence in the assessment results. In many cases various regulation bodies would require the tools used in the development to be certified to meet stringent quality requirements. The stiffness of an MC can make it difficult to meet this requirement. Careful selection of the method and tools used to solve accurately and efficiently stiff MCs is needed.

In the last 30 years, many approaches have been developed to deal efficiently with the MC stiffness [4, 5, 6, 7]. They can be split into two groups - "stiffness-tolerance" (STA) and "stiffness-avoidance" approaches (SAA) [5]. The main feature of STA is solving a stiff MC using *special numerical methods* that can provide highly accurate results. The limitations of STA are: i) STA cannot deal effectively with large models, and ii) computational efficiency is difficult to achieve when highly accurate solutions are sought. The SAA solution, on the other hand, is based on an *approximation algo-*

rithm which converts a stiff MC to a non-stiff chain first, which typically has a significantly smaller state space [4]. An advantage of this approach is that it can deal effectively with large stiff MCs. Achieving high accuracy with SAA, however, may be problematic.

A number of software tools have been developed and applied to solving models of complex systems such as SHARP, Save [8], Reno, λ Predict, Möbius, etc. Among them is the utility developed by some of the authors of this paper (ODU) developed more than 15 years ago which is based on EXPMETH [21] and has been validated extensively on a range of models [18 - 20]. The utility uses the algorithm of *modified exponential method*. In addition, a number of off-the-shelf mathematical software packages exist which can be used for solving Markov models, e.g. Maple (Maplesoft), Mathematica (Wolfram Research) and MATLAB (Mathworks) which use standard methods for solving differential equations. These math packages enjoy high reputation among the respective customers earned over several decades by providing a wide range of solutions and good support with regular updates.

In this paper we present an empirical study using two systems: i) a computer system with hardware redundancy and diverse software under the assumptions that the rate of failure of software may vary over time, and ii) a queuing system with a server break-down and repair [4]. The solution of the first system is based on the principle of multi-fragmentation [9], one of the efficient methods of solving an MC in case the model parameters change over time. The main idea of this principle is that the MC is represented as a set of fragments with *identical structure*, but which differ in the values of one or more parameters. Each of the systems included in the study is described by a stiff MC. The main difference between the two systems is that the ratio between the stiffness indices (to be defined below) of the first and the second system is 10^3 . In other words we chose them to be quite different in terms of their stiffness index so that we could study if the stiffness index impacts the accuracy of the different methods for solving the respective models. Both systems were solved using STA. The second system was also solved using SAA. Thus we could compare the accuracy of STA and SAA when applied to solving the same (the second) system and how these compare with the exact solution, which for the second system is available in [10], [11].

We report that indeed the stiffness index impacts significantly the accuracy of the solution methods. We also offer a selection procedure which allows one to choose (among the many available for solving stiff MCs) the solution method that provides the best accuracy given the value of the stiffness index of the MC to be solved. We provide also a justification for our recommendations based on the comparison of the different methods when applied to the chosen two systems described by stiff MCs.

The numerical transient analysis of MCs is faced with two computational difficulties – the model stiffness and largeness, which can affect the accuracy of the solutions obtained. Several numerical methods are widely used that address these difficulties, among them the Rosenbrock method [13], [14], the TR-BDF2 [5], [7], [14], the Jensen's method (uniformization) [5], the implicit Runge-Kutta method [5], [6], [12], and the modified Gir method [6].

The Rosenbrock method is the one-step numerical method that has the advantage of being relatively simple to understand and implement. For moderate accuracy (tolerances of order $10^{-4} - 10^{-5}$) and systems of moderate-size ($N \leq 10$) the method allows for obtaining solutions which in terms of achieved accuracy are comparable with the more complex algorithms. If a low accuracy is acceptable, then this method is attractive. When larger systems are solved the Rosenbrock method becomes less accurate and reliable [13].

TR-BDF2 is a second order accurate A-stable and L-stable single step composite method that uses one step of trapezoidal rule (TR) and one step of BDF2 (second order backward difference formula) [7]. [14] demonstrated that TR-BDF2 deals well with increased stiffness and only requires little extra computations as the parameter values or the mission time are increased. TR-BDF2 is also recommended for use if low accuracy is acceptable [5].

The Jensen method (also known as *uniformization* or randomization) [15] involves the computation of Poisson probabilities. It was extensively modified [5], [14], [16] to deal with the stiffness problem. It achieves greater accuracy than TR-BDF2 but still deals poorly with stiffness in extreme cases. [14] recommends that the Jensen method be used only in cases of moderately stiff models.

The implicit Runge-Kutta method is a single step numerical method that deals with the problem of stiffness and is one of the most computationally efficient methods for achieving high accuracy [6].

Also an aggregation/disaggregation technique for transient solution of stiff MCs was developed by K. S. Trivedi and A. Bobbio [4]. The technique can be applied to any MC, for which the transition rates can be grouped into *two separated sets of values*: one of the sets would include the “slow” states and the second set would include the “fast” states [4]. After aggregating the fast transition rates the MC is reduced to a smaller non stiff MC, which can be solved efficiently using a standard numerical technique [4].

In the rest of the paper the method by Trivedi and Bobbio [4] is referred to as an SAA while the other methods surveyed above are referred to as an STA.

The focus of this study is a comparison of the accuracy of the solution obtained with different methods when applied to the same system. Of particular interest is how the solutions are affected by the stiffness of the system under study.

The rest of the paper is organized as follow: in the section 2 we describe formally the stiffness problem and the stiffness index introducing informally the idea of how stiffness index may impact the accuracy of the numerical methods used in solving the systems. In section 3 we present the comparison results. In section 4 we present a procedure for selecting the optimal solution method and tool based on the stiffness index of an MC. In section 5 we present the conclusions and the problems left for future research.

2 Comparative Analysis of Evaluation Techniques

2.1 The Stiffness Problem

Stiffness is an undesirable property of many practical MCs that pose difficulties in finding transient solutions. There is no commonly adopted definition of “stiffness” but a few of the most widely used ones are summarized below.

The Cauchy problem $\frac{du}{dx} = F(x, u)$ is said to be stiff on the interval $[x_0, X]$ if for x from this interval the following condition is fulfilled:

$$s(x) = \frac{\max_{i=1,n} |\operatorname{Re}(\lambda_i)|}{\min_{i=1,n} |\operatorname{Re}(\lambda_i)|} \gg 1, \quad (1)$$

where the $s(x)$ – denotes the index of stiffness (stiffness index) and λ_i – are the eigenvalues of a Jacobian matrix ($\operatorname{Re} \lambda_i < 0, i = 1, 2, \dots, n$) [6].

Also the index of stiffness of an MC was defined in [5], [14] as the product of the largest total exit rate from a state and the length of solution interval ($=\lambda_i t$), where λ_i are the eigenvalues of the Jacobian matrix.

A system of differential equations (DE) is said to be stiff on the interval $[0, t]$ if there exists a solution component of the system that has variation on that interval that is large compared to $1/t$. Thus, the length of the solution interval also becomes a measure of stiffness [14], [17].

We use the index of stiffness in the empirical evaluations that follow as a measure of discrimination between the MCs with different indices of stiffness: high-stiffness ($s(x) \geq 10^3$), moderate-stiffness ($10^2 < s(x) < 10^3$) and low-stiffness ($s(x) \leq 10^2$).

Here we provide an illustration of how the index of stiffness can affect the accuracy achievable by numerical methods of solving a system of the DEs, which describes a stiff MC.

The most common type of a stiff linear system of DEs is the system in which the eigenvalues can be divided into two groups, based on the difference in their modulus values. The eigenvalues of the first group with large modulus values determine the solution behaviour in the boundary layer. Their corresponding components are rapidly decreasing. The eigenvalues of the second group with small modulus values determine the solution behaviour out of the boundary layer. The index of stiffness (1) is the ratio between the maximum value from the first group and the minimum value from the second one.

To describe in detail the influence of this separation on the stability of the numerical methods let us consider a DE matrix with constant coefficients,

$$y' = Ay, A = (a_{ij}), i, j = 1, \dots, M \quad (2)$$

$$y(0) = y_0, y_i = (y_0^i), i = 1, \dots, M \quad (3)$$

Matrix A is a simple-structured matrix, which means that it has M linearly independent eigenvectors and λ_i, e_i – are the eigenvalues and the corresponding eigenvectors of matrix A, respectively. The solution of the Cauchy problem (2) with initial conditions (3) is presented in (4):

$$y(x) = \sum_{i=1}^M C_i e^{\lambda_i x} e_i \tag{4}$$

Each component $C_i e^{\lambda_i x} e_i$ present in the solution (4) is proportional to one of the eigenvectors and is integrated independently of the other components.

If matrix A has large absolute negative eigenvalues a very small step h would be required on the whole integration interval. With a large integration interval this limitation would cause an increase of the local round-off errors, which would become a serious problem if high overall accuracy is sought.

As an example let's consider a system of two differential equations. Without loss of generality let us assume that $|\lambda_1| \gg |\lambda_2|$.

The exact solution (4) will take the following form:

$$y(x) = C_1 e^{\lambda_1 x} e_1 + C_2 e^{\lambda_2 x} e_2 \tag{5}$$

The first component of of the solution will decrease rapidly on the interval $\tau = 1/|\lambda_1|$, and after that will become extremely small. In this interval this component influences the solution $y(x)$. The second component changes on the interval $\tau = 1/|\lambda_2|$. The second interval is much wider than the first one. In this interval the second component influences the solution $y(x)$. In the first interval the rate of solution change is high and it is dominated by the change of the first component. In the second interval, the rate of change is small and is dominated by the change of the second component. As we can see the values of the given coefficients affect the behaviour of the transient solution. On the first interval, the so called *boundary layer*, in order to achieve an accurate solution in the presence of rapid changes, the step-size must satisfy the condition $h \ll 1/|\lambda_1|$. In the second interval the same condition on the step is required, because the corresponding component of the DE must decrease.

The example despite its simplicity provides a clear illustration of how different eigenvalues may lead to the need for changing the step-size in different integration intervals so that accurate results may be obtained. The value of the stiffness index (1) clearly affects the accuracy achievable with a given solution method. The higher the stiffness index the stricter the requirements imposed on the stability of the chosen numerical method.

We study in detail the impact of the stiffness index on the achievable accuracy with different numerical methods using two stiff MCs with substantially different stiffness indices, $s(x)$. The first stiff MC is a model of a computer system with hardware redundancy and diverse software (S₂₂) [21]. The second system is a queuing system with server break-down and repair (M/M/1/k) [5]. The first system is of moderate-stiffness, with $s(x) = 4 \cdot 10^2$ (1), where $\max |\text{Re}(\lambda_i)| = 0.2$ and $\min |\text{Re}(\lambda_i)| = 0.0005$. The MC of the second system is of high-stiffness, with $s(x) = 3.0001 \cdot 10^4$, where $\max |\text{Re}(\lambda_i)| = 3.0001$ and $\min |\text{Re}(\lambda_i)| = 0.0001$. Both MCs are of equal size – 20 states. The

study was conducted using the functions for solving stiff DEs implemented in the mathematical package MATLAB – *ode23s*, *ode23tb*, *ode15s*, which implement the Rosenbrock method, the TR-BDF2 and the method of backward differentiation, respectively.

Table 2. Experiment results

Method	Parameter	$t_1=[0,1000]$	
		S_{22}	M/1/M/m
Rosenbrock	NSS	93	172
	FE	2141	4302
	NLS	279	516
TR-BDF2	NSS	114	210
	FE	269	506
	NLS	361	692
Backward differentiation	NSS	74	150
	FE	111	203
	NLS	89	179

Table 1 summarizes the parameters obtained with the different methods for solving stiff DE: the number of successful steps (NSS), the function evaluations (FE) and the number of solutions of the linear systems (NLS). The time interval is $t_1=[0;1000]$. The Table shows that even when the model largeness is the same the solution for a system of high-stiffness, M/1/M/m, requires nearly twice as many steps, function evaluations and linear system solutions.

2.2 Stiffness-Tolerance and Stiffness-Avoidance Approaches

Stiffness-Tolerance Approach. The main idea of this approach is using methods that are stable for solving stiff models. These can be split into two broadly classes: “classical” numerical methods for solution of stiff DEs and “modified” numerical methods used for finding a solution in special cases.

(a) The classical (non-modified) numerical methods for solving stiff DEs use special single-step and multi-step integration methods. Examples of such methods are the implicit Runge-Kutta, the TR-BDF2, the Rosenbrock method, the exponential method, the implicit Gir method described in [5], [6], [7], [13], respectively. The implicit Runge-Kutta, TR-BDF2 and Rosenbrock method are implemented by several mathematical off-the-shelf software packages and are usually considered the most accurate methods for solving stiff ODEs.

(b) An example of the modified numerical methods is the exponential modified method. The original algorithm was presented in [6] and is based on the evaluation of the matrix exponent. In [6] this method is recommended as one of the most effective algorithms for solving the class of ODE systems with a high value of the Lipchitz constant, and as a special part of a stiff ODE. As a modification part, an automated adaptive step of integration can be implemented [6]. As the method has a multi-step

algorithm the given modification can increase the accuracy of the solution. The amount of computations and the machine time needed for the solution of stiff DEs can be reduced, too [6].

The solution provided by using any numerical method is expected to be accurate. However, typically the result obtained with numerical method include errors coming from different sources, such as: *problem statement error* - an inherent error, due to various simplifications introduced in order to make the problem (analytically) tractable; *truncation error* - the error related to truncating the infinite series after a finite number of terms are computed; *round-off error* - the type of error that arises in every arithmetic operation carried out on a computer; *initial error* - the error related to the presence of approximate parameters in the mathematical formulae. Ideally we would like to control each of these components of the computational error.

Stiffness-Avoidance Approach. The basic idea of this approach is a model transformation by identifying and eliminating the stiffness from the model, which would bring two benefits: i) a reduction of the largeness of the initial MC, and ii) efficiency in solving a non-stiff model using standard numerical methods. The approach was named an aggregation/disaggregation technique for transient solution of stiff MCs. The technique, developed by K. S. Trivedi, A. Bobbio and A. Reibmann [4], [11], can be applied to any MC with transition rates that can be grouped into two separate sets of values – the set of *slow* and the set of *fast* states [4].

While the transformation of the initial stiff MC brings benefits in terms of efficiency, to the best of our knowledge, no systematic study has been undertaken of the impact of the transformation (from a stiff to a non-stiff MC on the accuracy of the solution. In addition, since the method is not supported by standard off-the-shelf tools, the scope for human error in applying it is non-negligible.

3 Examples and Results

3.1 Example Systems Used in the Studies

In this subsection we provide a brief description of the systems that were solved using STA and SAA. The first system was solved using both approaches, while the second one – using only the SAA. The solution of the second system using STA was presented in [11, 12].

System 1: A Fault-Tolerant Computer System. The first system, Fig. 1, used in the study is a fault-tolerant computer system with two hardware channels, on which diverse control software is run.

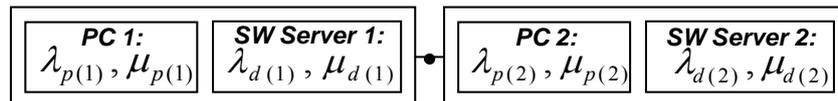


Fig. 1. Reliability block diagram of the chosen fault-tolerant system

In this case increasing system reliability is achieved via redundant hardware-software components with identical hardware structure that supports “hot backup” [21]. We assume that software run of the hardware channels is diverse [18], i.e. non-identical but functionally equivalent software copies are deployed on the hardware channels. The architecture thus offers protection against software design faults. Two channel configurations are very widely used in many safety-critical application, e.g. in instrumentation of nuclear plants, for instance Quad 3000 SIS critical control and safety application. Also similar architectures are used in many business-critical applications, such as the fault-tolerant servers (Blade Server NS50000c, IBM z10, Sun SPARC Enterprise M9000 [21]).

Informally, the operation of the system is as follows. Initially the system is working correctly – both hardware and software channels deliver the service as expected. If during operation one of the hardware channels has failed, the system operation will be failed over to the second channel until the first channel is “repaired”. Similarly, a software component may fail, in which case a failover will take place to the other channel, etc. In addition, we assume that the rates of failure and repair of software will vary over time, e.g. as a result of executing the software in partitions as discussed in [19]. We implement this assumption based on the research work [21]. This assumption captures a plausible phenomenon – variation of software failure rates - which is well accepted in practice: various software ‘aging effects’ are indeed modelled by an increased rate of software failure.

Model Parameters. The model parameters are as follows: *i*) $\lambda_{p(1)}$, $\mu_{p(1)}$ and $\lambda_{p(2)}$, $\mu_{p(2)}$ – hardware failure and repair rates of the first and second hardware channels, respectively; *ii*) $\lambda_{d(1)}$, $\lambda_{d(2)}$ – the initial software failure rate of the 1st and 2nd software versions; *iii*) $\Delta\lambda_d$ – the step of failure rate decrease after the software recovers from a failure; *iv*) $\mu_{d(1)}$ and $\mu_{d(2)}$ – the initial software repair rate of the 1st and the 2nd software versions; *v*) $\Delta\mu_d$ – the step of software repair rate decrease after the software recovers from a failure. We also assume that the values of system failure and repair rates of both the hardware components and of the software versions are equal: $\lambda_{p(1)} = \lambda_{p(2)}$, $\mu_{p(1)} = \mu_{p(2)}$, $\lambda_{d(1)} = \lambda_{d(2)}$, $\mu_{d(1)} = \mu_{d(2)}$ [21].

The Markov transition graph for the system presented in Fig. 1 is shown in Fig. 2. The model is built using the principle of multi-fragmentation [9]. Using this principle the model can be divided into N fragments that are with the same structure but may differ in one or more parameter values [21]. The number of fragments N in the MC depends on the number of expected undetected software faults N_d , the value of which can be estimated using probabilistic prediction models (6) [21]:

$$N = N_d + 1 \quad (6)$$

The sum of the failure rates of both software versions is defined as (7):

$$\Lambda_d = \lambda_{d(1)} + \lambda_{d(2)} \quad (7)$$

This MC of System 1 consists of the following states: $SF_1 = \{S_1, S_4, \dots, S_{n+1}\}$ – the set of states when both the hardware and software on both channels are working correctly, $SF_2 = \{S_2, S_5, \dots, S_{3n+2}\}$ – the set of states in which one of the hardware channel

has failed, $SF_3 = \{S_3, S_6, \dots, S_{3n+3}\}$ – the set of states in which one of the software versions has failed.

The system's operation is described next. At time t_0 the system operates correctly in S_1 . At random moment, t_n , a hardware or a software component failure occurs. In case of a hardware failure the system moves to state S_2 . The rate of this transition is $2\lambda_p$ and recovers from this state back to S_1 with rate μ_p . In case of software failure the system moves to state S_3 and recovers from this failure by moving to state S_4 with rate μ_d . The states S_1, S_2 and S_3 form the first fragment of the model, S_4, S_5 and S_6 – form fragment 2, etc. We assume that the internal fragments rates μ_d and Λ_d decrease by $\Delta\mu_d$ and $\Delta\lambda_d$, respectively, as the system moves between fragments from left to right.

From the (Fig. 2) we derive the matrix of the system transition rates (8), where $i=(1, \dots, n)$ is the number of system fragments:

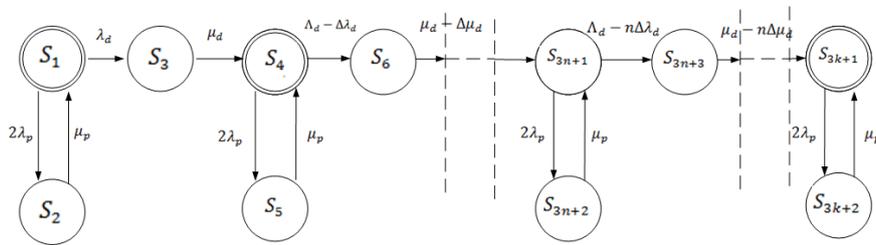


Fig. 2. Model of the system to be studied

$$\begin{pmatrix}
 -(2\lambda_p + \lambda_d) & \mu_p & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 2\lambda_p & -\mu_p & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 \lambda_d & 0 & -\mu_d & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & \mu_d & -(\Lambda_d - i\Delta\lambda_d + 2\lambda_p) & \mu_p & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 2\lambda_p & -\mu_p & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & \Lambda_d - i\Delta\lambda_d & 0 & -(\mu_d - i\Delta\mu_d) & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\
 \dots & \dots \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & -(\mu_d - n\Delta\mu_d) & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & \mu_d - n\Delta\mu_d & -(\Lambda_d - n\Delta\lambda_d + 2\lambda_p) & \mu_p & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 2\lambda_p & -\mu_p & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & \Lambda_d - n\Delta\lambda_d & 0 & -(\mu_d - n\Delta\mu_d) & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \mu_d - n\Delta\mu_d & -2\lambda_p & \mu_p \\
 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 2\lambda_p & -\mu_p
 \end{pmatrix} \tag{8}$$

System 2: A Queuing System with Server Breakdown and Repair. The second system was described in details by K.S. Trivedi and A. Bobbio in [4, 10]. The authors consider an M/M/1/k queuing system where m is the system capacity. The first “M” denotes a Poisson arrival process, the second “M” – denoted an exponentially distributed service times. The system has 1 server and k buffers to hold customers waiting for a service. The resulting model operates in 22 states [11]. Fig. 3 shows the Markov transition graph for the system.

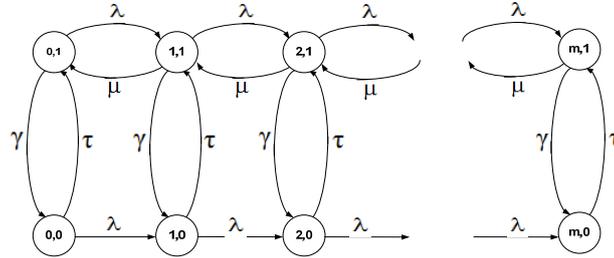


Fig. 3. The state diagram of the M/M/1/m queuing system with a server breakdown and repair

Model Parameters. The λ and μ are the arrival and the service rates, respectively, while γ and τ denote the server failure and failure repair rates. The main assumption is that the rates λ and μ are fast while γ and τ are slow [4]. Using the parameters presented in [11] we computed the index of stiffness for this system to be $s(x) = 3 \cdot 10^4$. Under the terms of this paper this system is classified as a *high-stiffness system*. The system was solved on the same interval $[0, 1000)$ [11].

3.2 Experimental Results

In this Subsection we present the results obtained using the approaches described in Section 2. The solutions for the MC of high-stiffness (the queuing system) is referred to as *Experiment 1*. The solution for the MC of moderate-stiffness (the fault-tolerant computer system) with changing parameters, would be referred to as *Experiment 2*. Lastly, we define an MC of low-stiffness: this is a variant of the fault-tolerant computer system with two hardware channels in which the model parameters have been changed so that the stiffness index has become low ($s(x) = 50$). This solution will be referred to as *Experiment 3*.

Experiment 1. A solution of the *queuing system with sever breakdown and repair* using SAA was presented in [4], [11]. In [4] as a measure of interest the authors used the expected number of customers in the queue, $E[N]$. In [11], in addition, the approximate result (calculated using SAA) and the exact values of $P_{22}(t)$ - the probability of having all buffers full and the server down – were plotted together.

Now we turn our attention to solve the MC of high-stiff using the STA to solve this model. We used two methods that fall into the STA category: using the EXPMETH utility; using the functions for solving stiff DEs implemented in the mathematical package Mathematica.

We used EXPMETH with initial data as follows: the matrix of coefficients was set as defined in [11]; the mission time was set to $t = 1000$; the accuracy of the solution sought was set to 10^{-6} ; the number of steps was set to $n = 20$. As Table 2 illustrates we obtained a negative result, where the probabilities were calculated for every 50 hours of the mission time, $S = \{S_1, S_2, S_3, \dots, S_{20}\}$.

Table 2. Results obtained for Experiment 1 with ODU (EXPMETH)

$t \backslash S_n$	S_1	...	S_{20}
50	1.76231646111739250e+0100- 4.15365304345735515e+0100	...	1.06048815978458797e+0095- 3.22465594380700072e+0094
...
1000	1.88981041657022775e+2054- 4.45414711917994213e+2054	...	1.13720867698699733e+2049- 3.45794216161554014e+2048

The results from using the functions for solving stiff DEs implemented in Mathematica are summarised in Fig. 4 in which the exact solution for $P_{22}(t)$ for $\lambda_1=1.0$, given in [11], is compared with the results obtained with Mathematica.

Based on the empirical evidence with the STA methods applied to *Experiment 1* we conclude that STA cannot provide highly accurate solution for MCs of high-stiffness.

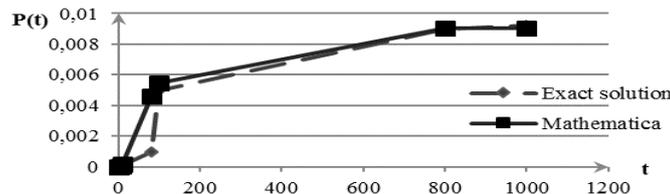


Fig. 4. Results of solving Experiment 1 model using Mathematica

Experiment 2. To solve a moderately-stiff MC (Experiment 2) using SAA we used the algorithm described in [4] and the uniformization method. STA solutions are also obtained using EXPMETH and the mathematical package Mathematica, respectively.

The mission time was set again to $t=1000$, $s(x)=4*10^2 (1)$, where $\max |\text{Re}(\lambda_i)|=0.2$ is the value of μ_d , the software repair rate in the initial model fragment and $\min |\text{Re}(\lambda_i)|=0.0005$ is the value of μ_d for the software repair rate in the final model fragment.

A comparison between the solutions is shown in Table 3 for values of the probabilities that the system is working in each of the states: $\{S_1, S_4, S_7, S_{10}, S_{13}, S_{16}, S_{19}\}$ at $t=500$. This set represents the states without failure (operational states, i.e. both channels work correctly without hardware or software failure).

Fig. 5 shows the results for system availability obtained with EXPMETH and Mathematica. For this system (Experiment 2) we used the result obtained with EXPMETH as an *exact solution* [18]. A discussion of the discrepancies between the solutions obtained with various packages is available in [18].

Based on this experiment we can conclude that for MCs of moderate - stiffness both the SAA and STA can be used. We note that the STA methods would produce an accurate solution faster than SAA when applied to small to moderate systems.

We also note that the particular mathematical package, Mathematica, detects automatically the stiffness of the DE's using a built in "StiffnessTest". In addition the user

of this package can use "StiffnessSwitching", the basic idea as the name suggests being that the package will switch automatically between stiff and non-stiff solvers depending on the outcome of the stiffness test. The non-stiff solver uses the "ExplicitModifiedMidpoint" base method, while the stiff solver uses the "LinearyImplicitEuler" base method [22]. Such special methods can be useful in case of solving an MC of small to moderate size. Finally, we note that the mathematical package offers convenience, but at same time significant effort is required to construct the necessary functions in case of large models, which introduces scope for human errors, e.g. while entering the initial data.

Table 3. Results comparison. System 1

State/t=500	SAA	STA	
		EXPMETH	Mathematica
$S_1(t)$	0,536010	0,537050	0,537052
$S_4(t)$	0,323950	0,321630	0,321634
$S_7(t)$	0,078610	0,078540	0,078542
$S_{10}(t)$	0,010180	0,009810	0,009814
$S_{13}(t)$	0,000640	0,000620	0,000618
$S_{16}(t)$	0,000021	0,000020	0,000023
$S_{19}(t)$	0	0	0

EXPMETH can be more effective in terms of usability. With *Experiment 2* Mathematica required a function with 7 arguments, while EXPMETH only required 4 arguments and a matrix of coefficients of the DEs.

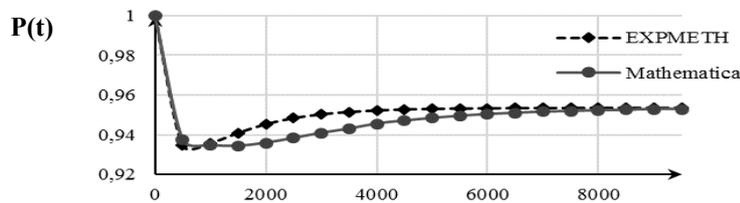


Fig. 5. Comparison of STA methods applied to Experiment 2

Experiment 3. We solved the model of a system with low-stiffness (*Experiment 3*) using the STA only. Based on the justification in Subsection 2.1 we concluded that SAA are the best for solving MCs of high-stiffness and large MCs of moderately-stiffness. In case of MCs of low-stiffness the use of STA can take less time and still provides an accurate solution. As in the previous experiment we consider a mission time $t=1000$, the $s(x)=50$ (1), where $\max |\operatorname{Re}(\lambda_i)|=0.2$ – the value of μ_d software repair rate in the initial model fragment and $\min |\operatorname{Re}(\lambda_i)|=0.004$ – the value of μ_d software repair rate in the final model fragment. Fig. 6 shows the results of the comparison of system availability, $P_a(t)$, obtained with EXPMETH and Mathematica. The

results are practically indistinguishable. Based on this experiment we can conclude that for MCs of low-stiffness the STA can provide an accurate result.

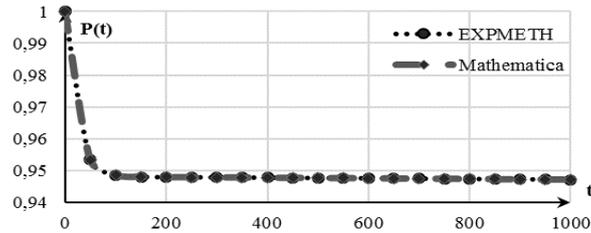


Fig. 6. Comparison of results with STA methods applied to Experiment 3

4 Selection of a Solution Method and of a Software Tool

Based on the results presented in the previous Subsection we propose the following selection procedure (Fig. 7), which takes into account the index of stiffness of the MC under consideration. The first layer of the algorithm takes the index of stiffness, $s(x)$ given by (1), as an initial separator. The system in question is assigned to one of the three classes: high-stiffness, moderate-stiffness or low-stiffness.

An MC of high-stiffness. If the value of $s(x)$ is greater or equal than 10^3 the system model is defined as an MC of high-stiffness. We move to the left branch of the algorithm. If in the system under consideration the parameters change over time, we propose that the principle of multi-fragmentation (MFM) be used. Otherwise this step is skipped. Based on the results from *Experiment 1*, we propose that SAA be used. In this case using specialized software will provide the most accurate solution. The use of STA in this case can produce a solution of low accuracy, which may be unacceptable.

An MC of moderate-stiffness. If the index of stiffness is in the interval $[10^2, 10^3]$ we move to the branch in the middle of the diagram. As in the previous case we propose that MFM be used if in the system under consideration the parameters vary over time. If the parameters do not change the procedure suggests that the initial MC (IMC) be used. On the third layer we propose that the largeness be used as an additional separator. As a theoretical separator the number of system states $n=1000$ can be used. If the number of model states is greater than n – the model is considered large, otherwise the model is not large. To provide effective results in case of a moderately-stiff large MC we propose the use of SAA and specialized tools. Indeed, one of the main features of SAA is the reduction of the system state space. For moderately-stiff MCs which are not large, based on the results of *Experiment 2*, we propose that STA be used with either EXPMETH or other specialized tools.

An MC of low-stiffness. If the index of stiffness is low, $s(x) \leq 10^2$, we move to the right branch of the algorithm. On the second layer we use the same separator as in previous cases. If the system under consideration includes parameter changes then MFM is needed, otherwise it is not needed and the initial MC can be used. In the third layer the system largeness is used as a separator. In case of a large MC of low-

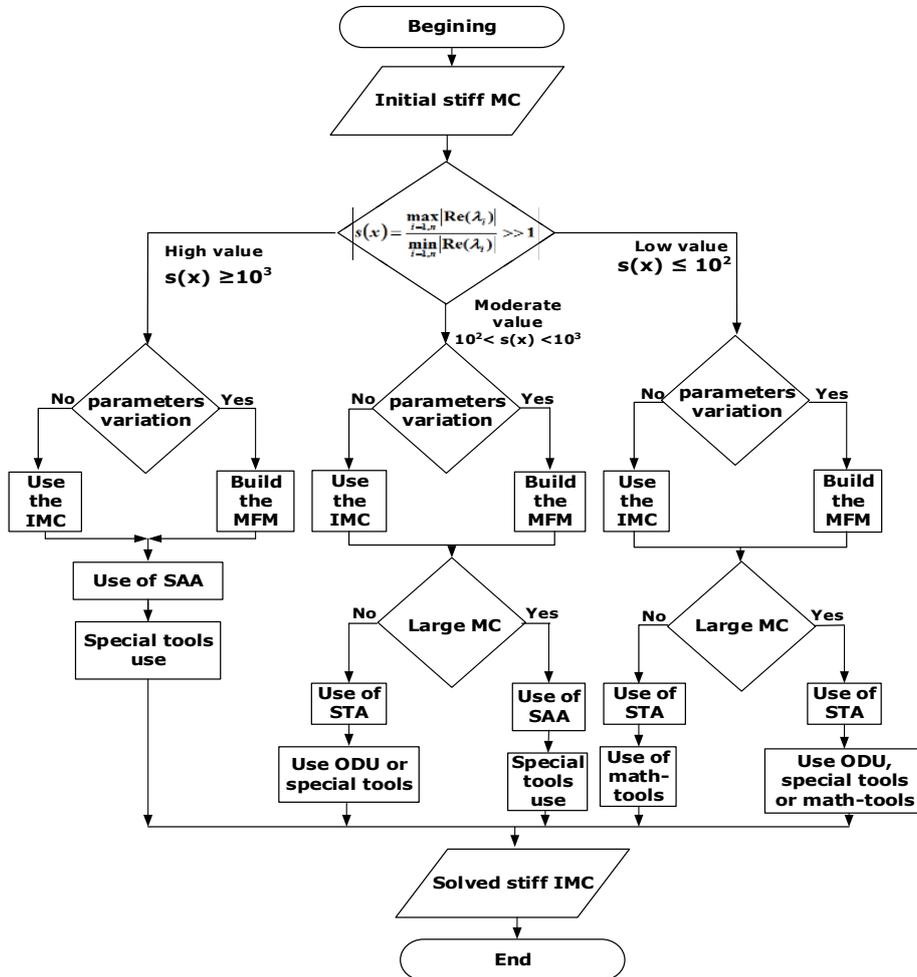


Fig. 7. An algorithm of selecting an optimal method for solving MCs based on the stiffness index and the size (largeness) of an MC

stiffness we propose that STA be used; as a tool we would recommend either EXPMETH or another specialized tool. These specialized tools were developed for special problems solution so they can provide more convenient data representation and satisfy the requirements of high accuracy. In the case of an MC of low-stiffness which is “not large” we propose the use of STA and EXPMETH or mathematical packages.

5 Conclusion

As a result of empirical studies we noticed that the value of stiffness index and the size of the MCs can affect the accuracy of the solutions achievable using different

methods. One of the interesting results is that we can effectively use the SAA to solve a large moderately-stiff MC when the parameters vary, which was the focus in previous research work [18]. In our future work we intend to extend the algorithm presented in the paper and take into account the most effective approach that can deal with large MCs: largeness-tolerant and largeness-avoidance approaches. As a result we are hoping to define the best combination of “largeness-stiffness” approaches that can be applied effectively to systems with variable parameters.

References

1. Volkov, L.: Managing the Operation of the Aircraft Systems: Tutorial. Vyshaya Shkola, Moscow (1981) (In Russian)
2. Ventsel', E., Ovcharov, L.: Probability Theory and its Applications in Engineering. Nauka, Moscow (2000) (In Russian)
3. Archana, S., Srinivasan, R., Trivedi, K. S.: Availability Models in Practice. In: Proc. Int. Workshop on Fault-Tolerant Control and Computing (FTCC-1), May 22-23, Seoul, Korea (2000)
4. Bobbio, A., Trivedi, K. S.: An Aggregation Technique for Transient Analysis of Stiff Markov Chains. IEEE Transactions on Computers, C-35, 803–814 (1986)
5. Malhotra, M., Muppala, J.K., Trivedi, K. S.: Stiffness-Tolerant Methods for Transient Analysis of Stiff Markov Chains. Microelectronic Reliability, 34(11), 1825–1841 (1994)
6. Arushanyan, O., Zaletkin, S.: Numerical Solution of Ordinary Differential Equations using FORTRAN. Moscow State University, Moscow (1990) (In Russian)
7. Bank, R. E. et al.: Transient Simulation of Silicon Devices and Circuits. IEEE Transactions on Electron Devices, 32(10), 1992–2007 (1985)
8. Geist, R., Trivedi, K. S.: Reliability Estimation of Fault-Tolerant Systems: Tools and Techniques. Computer. 23, 52–61 (1990)
9. Kharchenko, V., Timonkin, G., Sychev, V.: Fundamentals of Design and Constructions the Automated Control Systems for Aircraft Technical State Control. Study Guide. KVKIU, Kharkov (1992) (In Russian)
10. Nicola, V.F.: Markovian Models of Transactional System Supported by Check Pointing and Recovery Strategies. Part 1: a Model with State-Dependent Parameters. Eindhoven Univ. Technol., Eindhoven, the Netherlands, EUT Rep. 82-E-128 (1982)
11. Reibman A., Trivedi K. S., Kumar S., Ciardo G.: Analysis of Stiff Markov Chains. ORSA Journal on Computing, 1(2), 126–133 (1989)
12. Hayrer, E., Vanner, G.: Solution of Ordinary Differential Equations. Stiff and Differential-Algebraic Problems. Mir, Moscow (1999) (In Russian)
13. Press, W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P.: Numerical Recipes. The Art of Scientific Computing. 3d edition. Cambridge University Press (2007)
14. Reibman, A., Trivedi, K.S.: Numerical Transient Analysis of Markov models. Comput. Opns. Res., 15(1), 19–36 (1988)
15. Jensen A.: Markoff Chains as an Aid in the Study of Markoff Processes. Skand. Aktuarietidskrift, 36, 87–91 (1953)
16. Fox, B. L., Glynn, P. W.: Computing Poisson probabilities. Commun, ACM 31(4), 440–445 (1985)
17. Miranker, L.: Numerical Methods for Stiff Equations and Singular Perturbation Problems. Dordrecht, Holland (1981)

18. Kharchenko, V., Popov, P., Odarushchenko, O., Zhadan, V.: Empirical Evaluation of Accuracy of Mathematical Software Used for Availability Assessment of Fault-Tolerant Computer Systems. *RT&A* #03(26), 7, 85–97 (2012)
19. Littlewood, B., Popov, P., Strigini, L.: Modelling Software Design Diversity – a Review. *ACM Computing Surveys*, 33(1), 177–208 (2001)
20. Popov, P., Manno, G.: The Effect of Correlated Failure Rates on Reliability of Continuous Time 1-Out-of-2 Software. In: *Proc. Computer Safety, Reliability, and Security (SAFECOMP 2011)*, Naples, Italy (2011)
21. Kharchenko, V., Odarushchenko, O., Ponochovny, Y., Zhivilo, S., Odarushchenko, E., Kharibin, O., Odarushchenko, V.: High Availability Systems and Technologies. In: Kharchenko, V. (ed.). *Lectures, National Aerospace University “KhAI”* (2012)
22. Wolfram Mathematica 9 Documentation Center, <http://reference.wolfram.com/mathematica/tutorial/NDSolveStiffnessSwitching.html>

Asymptotical Information Bound of Consecutive Qubit Binary Testing

Anastasiia Varava and Grygoriy Zholtkevych

V.N. Karazin Kharkiv National University
School of Mathematics and Mechanics, 4, Svobody Sqr., 61022, Kharkiv, Ukraine

{nastia.varava,g.zholtkevych}@gmail.com

Abstract. The problem of estimating quantity of information, which can be obtained in the process of binary consecutive measuring a qubit state, is considered in the paper. The studied quantity is expressed as Shannon mutual information between the initial qubit state and the outcomes of a consecutive measurement. It is demonstrated that the maximum of information is reached by projective measurements. The maximum quantity of accessible information is calculated. It is proved that in the case of arbitrary binary test the maximum is reached asymptotically for consecutive measurements.

Keywords. accessible classical information, Shannon mutual information, pure qubit state, consecutive qubit binary testing

Key terms. MathematicalModel, MathematicalModelling, Research

1 Introduction

Industrial applications of quantum communication channels require developing sound method for constructing them. It is well known that Shannon information theory is a mathematical background for the such methods and the notion of Shannon mutual information [1, 9] is a basic instrument for the classical theory of communication. Thus, studying informational quantities for quantum communication channels is a very important problem for building quantum information theory.

The amount of accessible information on the quantum system is an important information-theoretic quantity. This information is obtained by performing quantum measurements on the given system. To study the possibility of obtaining information it is suitable to express it as Shannon mutual information between the qubit state and the outcomes of a measurement.

One of the first investigated problems with respect to the discussed quantity was the so-called problem of distinguishing quantum states. It consists of the following: suppose that quantum system is prepared in a state described by one of the density operators ρ_i ($i = 1, \dots, n$) with probability p_i . The goal is to determine which state is given using a single measurement of the considered

quantum system. Thus, the task is to find measurements providing the maximum of Shannon mutual information. This problem was investigated by Holevo [4], Davies [2] and many others. Particularly, Holevo proved in [4] that obtainable information I is less or equal than the so-called Holevo bound

$$I \leq S\left(\sum_i p_i \rho_i\right) - \sum_i p_i S(\rho_i),$$

where $S(\rho) = \text{Tr}(\rho \log \rho)$ is the von Neumann entropy. It follows that the amount of information obtainable by a single measurement never exceeds $\log(\dim \mathcal{H})$, where \mathcal{H} is a state space of a quantum system.

Consider a more general problem. Imagine that we do not have any preliminary information on the possible states of a quantum system. In this case all possible pure states are equiprobable, so, we can not work with a finite set of initial states. Assume also that we have in our disposal a single measuring instrument. How much classical information can we now obtain?

In the paper we consider the simplest case of this problem. Suppose we have an arbitrary pure qubit state and an instrument doing binary test of it. Assume that all possible initial states are equiprobable. Our aim is to obtain classical information on the qubit state using only the given measuring instrument. Certainly, we want to get as much classical information as possible. So, this time we investigate the amount of information accessible in this case.

It is well known that in the case of performing a single measurement the maximum of classical information is obtainable by a projective measurement. We present a simple proof of this fact. In addition, we calculate the maximum value of information and then we show that in the case of arbitrary measuring instrument this value can be attained asymptotically using consecutive testing.

This paper is organized as follows: in Section 2 the problem is set formally and the formula for Shannon mutual information of two random variables is obtained. The first one describes the density of the parameter of the initial state, and the second one corresponds to the measurement result. In Section 3 a special kind of measuring instrument which performs a projective measurement is considered. It is demonstrated that in this case we obtain the maximum of accessible information. Further, in Section 4, it is proved that in general case consecutive measurements provide to attain this value asymptotically.

2 Shannon Mutual Information between the Qubit State and the Outcomes of a Consecutive Measurement

To set the problem formally we firstly need some basic definitions.

In quantum computing, a qubit is the quantum analogue of the classical bit. Like a bit, a qubit have two basis states: $|0\rangle$ and $|1\rangle$. The difference is that a qubit can be in a superposition of the basis states at the same time. This means that a pure qubit state can be represented as a linear combination of $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

where α and β are probability amplitudes and can in general both be complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$.

We use a density operator as a mathematical model of a qubit state.

Definition 1. Let \mathcal{H}_n be an n -dimensional Hilbert space of a quantum system with n levels. A nonnegative definite operator ρ on \mathcal{H}_n is called a density operator if it holds the condition $\text{Tr}(\rho) = 1$.

Density operator ρ corresponds to a pure state if it is an one-dimensional ortho-projector. In other cases ρ corresponds to a mixed state.

We describe the considering pure state of a qubit by the corresponding density operator $\rho(\theta, \alpha)$. It can be represented by the following matrix

$$\rho(\theta, \alpha) = \begin{pmatrix} \frac{1}{2} - \theta & e^{i\alpha} \sqrt{\frac{1}{4} - \theta^2} \\ e^{-i\alpha} \sqrt{\frac{1}{4} - \theta^2} & \frac{1}{2} + \theta \end{pmatrix},$$

where parameters θ and α satisfy the conditions $-\frac{1}{2} \leq \theta \leq \frac{1}{2}$, $0 \leq \alpha < 2\pi$.

Let Θ be the equiprobability distribution on the segment $[-\frac{1}{2}, \frac{1}{2}]$. It is used to model uncertainty of our knowledge about parameter θ for an initial state of the qubit.

Let us describe the given measuring instrument using the following definition of a qubit binary test [5].

Definition 2. A pair of nonnegative definite operators $\mathbf{T} = \{M_0, M_1\}$ on \mathcal{H}_2 such that $M_0 + M_1 = \mathbf{1}$ is called a qubit binary test.

The mathematical model of a qubit binary test is described and studied in [10].

Let \mathcal{M} be a set of all possible qubit binary tests. Consider $\mathbf{T} \in \mathcal{M}$, $\mathbf{T} = \{M_0, M_1\}$. Let $\{|0\rangle, |1\rangle\}$ be the ortho-normal basis in \mathcal{H}_2 such that $|0\rangle, |1\rangle$ are eigenvectors of the operator M_0 corresponding to eigenvalues $0 \leq m_1 \leq m_2 \leq 1$ respectively. In this case the operators M_0 and M_1 are represented in a such way:

$$M_0 = \begin{pmatrix} m_1 & 0 \\ 0 & m_2 \end{pmatrix}, \quad M_1 = \begin{pmatrix} 1 - m_1 & 0 \\ 0 & 1 - m_2 \end{pmatrix}.$$

Denote by $\Sigma = \{0, 1\}$ the set of outcomes for the given qubit binary test. The probability distribution on this set is defined by the following formulae:

$$\Pr(0 | \rho) = \text{Tr}(\rho \cdot M_0), \quad \Pr(1 | \rho) = \text{Tr}(\rho \cdot M_1).$$

As we already mentioned, our aim is to obtain classical information value of the initial state as much as possible, using only the given instrument to measure. For this purpose we perform consecutive qubit binary testing. In other words, we repeat our measurement n times to observe the result of each iteration. After n

iterations we obtain a sequence $\mathbf{x}^{(n)} = (x_1, \dots, x_n)$, where $x_i \in \Sigma (i = 1, \dots, n)$. As it is shown in [10],

$$\Pr(\mathbf{x}^{(n)} \mid \rho(\theta, \alpha)) = m_1^k (1 - m_1)^{n-k} \left(\frac{1}{2} - \theta \right) + m_2^k (1 - m_2)^{n-k} \left(\frac{1}{2} + \theta \right),$$

where k is a number of '1' in the outcomes sequence $\mathbf{x}^{(n)}$.

In further computations it is suitable to use some properties of Bernstein basis polynomials of degree n [7]:

$$B_{n,k}(z) = \binom{n}{k} z^k (1 - z)^{n-k},$$

so, we rewrite the last equation as follows:

$$\Pr(\mathbf{x}^{(n)} \mid \rho(\theta, \alpha)) = \binom{n}{k}^{-1} \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right).$$

Let $\mathbf{X}^{(n)} \in \{0, 1\}^n$ be a random variable describing an outcomes sequence. The density of $\mathbf{X}^{(n)}$ with respect to Θ is defined by the following formula

$$p(\mathbf{x}^{(n)} \mid \theta) = \Pr(\mathbf{x}^{(n)} \mid \rho(\theta, \alpha)).$$

The main objective of this section is to compute Shannon mutual information [1, 9] of random variables $\mathbf{X}^{(n)}$ and Θ . It is equal to

$$I(\mathbf{X}^{(n)}; \Theta) = H(\mathbf{X}^{(n)}) - H(\mathbf{X}^{(n)} \mid \Theta), \quad (1)$$

where $H(\mathbf{X}^{(n)})$ is Shannon entropy of $\mathbf{X}^{(n)}$, and $H(\mathbf{X}^{(n)} \mid \Theta)$ is conditional entropy [1, 9]:

$$H(\mathbf{X}^{(n)} \mid \Theta) = \int_{-\frac{1}{2}}^{\frac{1}{2}} H(\mathbf{X}^{(n)} \mid \Theta = \theta) d\theta.$$

Let us compute the marginal distribution of the random variable $\mathbf{X}^{(n)}$, which corresponds to the joint density function $p(\mathbf{x}^{(n)}, \theta)$. The latter is described in a such way:

$$p(\mathbf{x}^{(n)}, \theta) = \binom{n}{k}^{-1} \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right).$$

The marginal distribution $p(\mathbf{x}^{(n)})$ is defined as follows:

$$\begin{aligned} p(\mathbf{x}^{(n)}) &= \int_{-\frac{1}{2}}^{\frac{1}{2}} p(\mathbf{x}^{(n)}, \theta) d\theta = \binom{n}{k}^{-1} \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \int_{-\frac{1}{2}}^{\frac{1}{2}} d\theta - \right. \\ &\quad \left. (B_{n,k}(m_1) - B_{n,k}(m_2)) \int_{-\frac{1}{2}}^{\frac{1}{2}} \theta d\theta \right) = \binom{n}{k}^{-1} \frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2}. \end{aligned}$$

Firstly, let us compute entropy of $\mathbf{X}^{(n)}$:

$$H(\mathbf{X}^{(n)}) = - \sum_{\mathbf{x}^{(n)} \in \{0,1\}^n} p(\mathbf{x}^{(n)}) \log p(\mathbf{x}^{(n)}) = - \sum_{k=0}^n \binom{n}{k} p(\mathbf{x}^{(n)}) \log p(\mathbf{x}^{(n)}) .$$

Substituting the expression of marginal distribution, we obtain

$$H(\mathbf{X}^{(n)}) = \sum_{k=0}^n \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \log \binom{n}{k} - \sum_{k=0}^n \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \cdot \log \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) . \quad (2)$$

Secondly, we can compute $H(\mathbf{X}^{(n)}|\Theta)$:

$$H(\mathbf{X}^{(n)}|\Theta) = - \int_{-\frac{1}{2}}^{\frac{1}{2}} H(\mathbf{X}^{(n)}|\Theta = \theta) d\theta = - \int_{-\frac{1}{2}}^{\frac{1}{2}} \sum_{k=0}^n \binom{n}{k} p(\mathbf{x}^{(n)}|\theta) \log p(\mathbf{x}^{(n)}|\theta) d\theta .$$

By direct calculations it is easy to check that

$$H(\mathbf{X}^{(n)}|\Theta) = \sum_{k=0}^n \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \log \binom{n}{k} - \sum_{k=0}^n \int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) \cdot \log \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) d\theta . \quad (3)$$

Now, using (1), (2) and (3), one can obtain the expression for Shannon mutual information

$$I(\mathbf{X}^{(n)}; \Theta) = \sum_{k=0}^n \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) \cdot \log \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) d\theta - \frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \cdot \log \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \right) .$$

It is easy to see that if k is such that $B_{n,k}(m_1) = B_{n,k}(m_2)$, then the corresponding summand is equal to zero:

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} (B_{n,k}(m_1) \cdot \log B_{n,k}(m_1)) d\theta - B_{n,k}(m_1) \cdot \log B_{n,k}(m_1) = 0 .$$

Note that in the case of equality $m_1 = m_2$ the polynomials $B_{n,k}(m_1)$ and $B_{n,k}(m_2)$ are equal for each k , so, we can not obtain any information about the initial state. Thus we further consider the case $m_1 < m_2$.

Let \mathbf{A}_n be a set of non-negative integers defined as follows:

$$\mathbf{A}_n = \{k \in \{0, \dots, n\} | B_{n,k}(m_1) \neq B_{n,k}(m_2)\}.$$

Denote by $\overline{\mathbf{A}_n}$ the complement of the set \mathbf{A}_n , i.e. $\overline{\mathbf{A}_n} = \{0, \dots, n\} \setminus \mathbf{A}_n$. Now we can consider only values of k from this set, but we keep on working with all summands. We demonstrate further that it is more suitable. Instead of omitting zero summands, let us present them in the following way:

$$0 = \left(B_{n,k}(m_1) - \frac{B_{n,k}(m_1)}{2 \ln(2)} \right) - \frac{2 \ln 2 - 1}{2 \ln 2} \cdot B_{n,k}(m_1).$$

Thus, we have

$$\begin{aligned} I(\mathbf{X}^{(n)}; \Theta) &= \sum_{k \in \mathbf{A}_n} \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) \cdot \right. \\ &\quad \log \left(\left(\frac{1}{2} - \theta \right) B_{n,k}(m_1) + \left(\frac{1}{2} + \theta \right) B_{n,k}(m_2) \right) d\theta - \\ &\quad \left. \frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \cdot \log \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \right) + \\ &\quad \sum_{k \in \overline{\mathbf{A}_n}} \left(\left(B_{n,k}(m_1) - \frac{B_{n,k}(m_1)}{2 \ln(2)} \right) - \frac{2 \ln 2 - 1}{2 \ln 2} \cdot B_{n,k}(m_1) \right). \end{aligned}$$

After integration we obtain

$$\begin{aligned} I(\mathbf{X}^{(n)}; \Theta) &= \sum_{k \in \mathbf{A}_n} \left(-\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{4 \ln 2} + \right. \\ &\quad \left. \frac{B_{n,k}(m_2)^2 \log B_{n,k}(m_2) - B_{n,k}(m_1)^2 \log B_{n,k}(m_1)}{2(B_{n,k}(m_2) - B_{n,k}(m_1))} - \right. \\ &\quad \left. \frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \cdot \log \left(\frac{B_{n,k}(m_1) + B_{n,k}(m_2)}{2} \right) \right) + \\ &\quad \sum_{k \in \overline{\mathbf{A}_n}} \left(\left(B_{n,k}(m_1) - \frac{B_{n,k}(m_1)}{2 \ln(2)} \right) - \frac{2 \ln 2 - 1}{2 \ln 2} \cdot B_{n,k}(m_1) \right). \end{aligned}$$

Taking into account that for k from the set $\overline{\mathbf{A}_n}$ the equality $B_{n,k}(m_1) = B_{n,k}(m_2)$ is held, we can rewrite this formula as follows:

$$I(\mathbf{X}^{(n)}; \Theta) = \sum_{k=0}^n \frac{2 \ln 2 - 1}{4 \ln 2} (B_{n,k}(m_1) + B_{n,k}(m_2)) - \sum_{k \in \overline{\mathbf{A}_n}} \frac{2 \ln 2 - 1}{2 \ln 2} B_{n,k}(m_1) + \sum_{k \in \mathbf{A}_n} \left(\frac{B_{n,k}(m_2)^2 \log B_{n,k}(m_2) - B_{n,k}(m_1)^2 \log B_{n,k}(m_1)}{2(B_{n,k}(m_2) - B_{n,k}(m_1))} - \frac{(B_{n,k}(m_1) + B_{n,k}(m_2)) \cdot \log (B_{n,k}(m_1) + B_{n,k}(m_2))}{2} \right).$$

Simplifying this expression and using the evident equality, $\sum_{k=0}^n B_{n,k}(z) = 1$, we get the following expression for mutual information:

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} - \sum_{k \in \overline{\mathbf{A}_n}} \frac{2 \ln 2 - 1}{2 \ln 2} B_{n,k}(m_1) + \frac{1}{2} \cdot \sum_{k \in \mathbf{A}_n} \left(\frac{B_{n,k}(m_2)^2 \log B_{n,k}(m_2) - B_{n,k}(m_1)^2 \log B_{n,k}(m_1)}{(B_{n,k}(m_2) - B_{n,k}(m_1))} - \frac{(B_{n,k}(m_2)^2 - B_{n,k}(m_1)^2) \cdot \log (B_{n,k}(m_1) + B_{n,k}(m_2))}{(B_{n,k}(m_2) - B_{n,k}(m_1))} \right). \quad (4)$$

3 Extremal Property of Projective Measurements

In this section we consider a special kind of measurement. Let $m_1 = 0$ and $m_2 = 1$. In this case the qubit binary test $\mathbf{T} = \{M_0, M_1\}$ looks as follows:

$$M_0 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad M_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

The first interesting property of this measurement is its repeatability. Actually, note that $M_i \cdot M_j = \delta_{i,j} \cdot M_i$. So, M_0 and M_1 are orthoprojectors, and \mathbf{T} is a projective measurement. Thus, in this case the repeated measurements give the same result. We demonstrate further that due to this property consecutive testing do not provide any extra information. The second and the most remarkable property is that we can obtain the maximum of the accessible information if and only if we use a projective measurement.

First of all let us compute the amount of information obtainable by the considering measurement. In the considered case we have only two values of k such that the corresponding polynomials $B_{n,k}(m_1)$ and $B_{n,k}(m_2)$ are not equal. So, $\mathbf{A}_n = \{0, n\}$, and

$$\forall k \in \overline{\mathbf{A}_n} : B_{n,k}(m_1) = B_{n,k}(m_2) = 0.$$

Now using (4) it is easy to see that considering $m_1 = 0$ and $m_2 = 1$ we obtain

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} .$$

Our next goal is to show that the obtained amount of information can not be reached using any other qubit binary test. For this purpose we investigate the function describing the accessible information (4).

At first let us rewrite this function in a more suitable way. Consider the general case, in which $0 < m_1 < m_2 < 1$. We will demonstrate further that the exceptional cases, when $0 = m_1 < m_2 < 1$ or $0 < m_1 < m_2 = 1$, are similar to the latter. Taking into account that in the general case $\forall k \in \{0, \dots, n\} B_{n,k}(m_1) > 0$, let us denote the ratio between $B_{n,k}(m_2)$ and $B_{n,k}(m_1)$ by a new function:

$$t_{n,k}(m_1, m_2) = \frac{B_{n,k}(m_2)}{B_{n,k}(m_1)} .$$

Thus, by direct calculations we obtain the following formula for mutual information:

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} - \frac{2 \ln 2 - 1}{2 \ln 2} \sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) - \frac{1}{2} \sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot \left(\frac{t_{n,k}(m_1, m_2)^2 \cdot \log \left(\frac{t_{n,k}(m_1, m_2)}{t_{n,k}(m_1, m_2) + 1} \right) + \log(t_{n,k}(m_1, m_2) + 1)}{1 - t_{n,k}(m_1, m_2)} \right) . \quad (5)$$

Let $f(t)$ be a function defined as follows:

$$f(t) = \frac{t^2 \cdot \log \left(\frac{t}{t + 1} \right) + \log(t + 1)}{1 - t} .$$

Now it is easy to see that the formula (5) can be written as

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} - \left(\frac{2 \ln 2 - 1}{2 \ln 2} \sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) + \frac{1}{2} \sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) \right) . \quad (6)$$

We claim that in the considered case ($0 < m_1 < m_2 < 1$) mutual information is less than

$$\frac{2 \ln 2 - 1}{2 \ln 2} .$$

To prove this fact it is enough to notice that the variable part of expression (6) is always negative. Actually, we know that $B_{n,k}(m_1) > 0$ and $t_{n,k}(m_1, m_2)$ is a

ratio of two positive values, so, it is also positive. In addition, it is easy to show in the classical way that for all $t \in (0, 1) \cup (1, \infty)$ function $f(t)$ is greater than zero.

Now we need to consider the special case, in which $0 < m_1 < 1$, $m_2 = 1$. It is evident that the case when $m_1 = 0$, $0 < m_2 < 1$ is similar to the latter.

Suppose that $0 < m_1 < 1$, $m_2 = 1$. Thus, on the one hand, for all k we have $B_{n,k}(m_1) > 0$. On the other hand, for all $k < n$ $B_{n,k}(m_2) = 0$, and only for $k = n$ $B_{n,k}(m_2) = 1$. It is easy to see that in this case we obtain

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} + \frac{1}{2} \cdot \left(\frac{B_{n,n}(m_2)^2 \log B_{n,n}(m_2) - (B_{n,n}(m_2)^2 - 1) \cdot \log(1 + B_{n,n}(m_2))}{B_{n,n}(m_2) - 1} \right) = \frac{2 \ln 2 - 1}{2 \ln 2} - \frac{1}{2} \cdot f(B_{n,n}(m_2)) < \frac{2 \ln 2 - 1}{2 \ln 2}.$$

Thus, now we know that

$$I(\mathbf{X}^{(n)}; \Theta) \leq \frac{2 \ln 2 - 1}{2 \ln 2},$$

and, in particular, the equality is held if and only if the considered binary test is projective.

4 Asymptotic Properties of Consecutive Measurements

In the previous section we have considered the case when the given qubit binary test is a projective measurement. We have proved that only this type of measurement allows to achieve the maximum of information about the initial state. As far as the measurement is projective, repeating of the measuring procedure does not provide any extra information. In addition, we have found the maximum value of the accessible information:

$$\max_{\mathbf{T} \in \mathcal{M}, n \in \mathbb{N}} \{I(\mathbf{X}^{(n)}; \Theta)\} = \frac{2 \ln 2 - 1}{2 \ln 2}.$$

In this section we return to considering of the general view of a qubit test, and we work with consecutive qubit testing. So, this time we investigate the dependence of the amount of information on n – the number of iterations. The objective of this section is to prove that the maximum of accessible information can be reached asymptotically by performing consecutive measurements using an arbitrary qubit binary test.

More strictly, our aim is to prove the next theorem:

Theorem 1. *Suppose we have a pure qubit state and we perform consecutive qubit binary testing using the given test $\mathbf{T} = \{M_1, M_2\}$. Then for arbitrary*

$\varepsilon > 0$ there exists a corresponding number of iterations $n(\varepsilon)$ such that for all subsequent iterations ($n > n(\varepsilon)$) the following inequality is held:

$$\max_{\mathbf{T} \in \mathcal{M}, m \in \mathbb{N}} \{I(\mathbf{X}^{(\mathbf{m})}; \Theta)\} - I(\mathbf{X}^{(\mathbf{n})}; \Theta) < \varepsilon .$$

In other words, as far as the mutual information can be written as

$$I(\mathbf{X}^{(\mathbf{n})}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} - \left(\frac{2 \ln 2 - 1}{2 \ln 2} \sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) + \frac{1}{2} \sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) \right) ,$$

we need to find $n_1(\varepsilon)$ such that for all $n > n_1(\varepsilon)$

$$\sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{2} , \quad (7)$$

and $n_2(\varepsilon)$ such that for all $n > n_2(\varepsilon)$

$$\sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) < \frac{\varepsilon}{2} . \quad (8)$$

Therefore, for $n > n(\varepsilon) = \max\{n_1(\varepsilon), n_2(\varepsilon)\}$ both of these inequalities are held.

Let us fix a certain positive value of ε . At first we consider the left side of inequality (7). Let us divide the set \mathbf{A}_n into two non-intersecting subsets:

$$\begin{aligned} \Gamma_n(m_1) &= \{k \in \mathbf{A}_n\} : \left| \frac{k}{n} - m_1 \right| < \tilde{\delta}(m_1, m_2) , \\ \Delta_n(m_1) &= \{k \in \mathbf{A}_n\} : \left| \frac{k}{n} - m_1 \right| \geq \tilde{\delta}(m_1, m_2) , \end{aligned}$$

where $\tilde{\delta}(m_1, m_2)$ is a certain positive function.

It was demonstrated in [7], that for $0 < m_1 < 1$ and $\delta > 0$:

$$\sum_{k \in \Delta_n(m_1)} B_{n,k}(m_1) \leq \frac{1}{4n\delta^2} . \quad (9)$$

On the one hand, it is easy to see that $f(t)$ is a bounded function. Suppose that for all $t \in (0, 1) \cup (1, \infty)$ $f(t) < C$, where C is a certain positive constant. Thus we have

$$\sum_{k \in \Delta_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) \leq \frac{C}{4n\tilde{\delta}^2(m_1, m_2)} .$$

So, we can choose a value $n_{1,1}(\varepsilon)$ such that for all $n > n_{1,1}(\varepsilon)$

$$\sum_{k \in \Delta_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{4} .$$

On the other hand, we can see that when k is close to $n \cdot m_1$, the value of $t_{n,k}(m_1, m_2)$ goes to zero as n goes to infinity. As far as $\lim_{t \rightarrow 0} f(t) = 0$, there exists a value $n_{1,2}(\varepsilon)$ such that for all $n > n_{1,2}(\varepsilon)$

$$\sum_{k \in \Gamma_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{4}.$$

Now let $n_1(\varepsilon)$ be a maximum of values $n_{1,1}(\varepsilon)$ and $n_{1,2}(\varepsilon)$. Thus, for all $n > n_1(\varepsilon)$ inequality (7) is held.

Finally, let us consider inequality (8). Note that in the case of inequality $m_1 < m_2$ the set \mathbf{A}_n contains at most one element. Actually, by construction $k \in \mathbf{A}_n$ if and only if $B_{n,k}(m_1) = B_{n,k}(m_2)$. Solving this equation for the variable k , we have

$$k_0 = n \cdot \frac{\ln((1 - m_1)/(1 - m_2))}{\ln(((1 - m_1) \cdot m_2)/((1 - m_2) \cdot m_1))}.$$

If n , m_1 and m_2 are such that k_0 is an integer then $\overline{\mathbf{A}_n} = \{k_0\}$. If not, the set $\overline{\mathbf{A}_n}$ is empty. It is easy to show that $\lim_{n \rightarrow \infty} B_{n,k}(m_1) = 0$, so, we can easily find $n_2(\varepsilon)$ such that for all $n > n_2(\varepsilon)$ inequality (8) is held.

Now let us build a rigorous proof of the considering statement using this heuristic consideration. To do it, we firstly need several trivial propositions.

Proposition 1. *The following statements are correct:*

1. for all $x \in (0, 1)$ the inequality $x > \ln(x + 1)$ is true;
2. for all $x > 0$ the inequality $\ln(x/(x + 1)) < -1/(x + 1)$ is true too.

This proposition can be easily proved using Taylor series expansion of $\ln(x)$ and Euler's transform applied to this expansion.

Proposition 2. *Let $x, y \in (0, 1)$ and $x \neq y$ then the following inequality is held:*

$$\left(\frac{x}{y}\right)^y \left(\frac{1-x}{1-y}\right)^{(1-y)} < 1.$$

The proof is omitted.

Now we can prove the above formulated theorem.

Proof (of Theorem 1). As we already know, the mutual information can be presented as

$$I(\mathbf{X}^{(n)}; \Theta) = \frac{2 \ln 2 - 1}{2 \ln 2} - \left(\frac{2 \ln 2 - 1}{2 \ln 2} \sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) + \frac{1}{2} \sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) \right).$$

We also know that

$$\max_{\mathbf{T} \in \mathcal{M}, n \in \mathbb{N}} \{I(\mathbf{X}^{(n)}; \Theta)\} = \frac{2 \ln 2 - 1}{2 \ln 2}.$$

To prove the theorem it is enough to show that there exists $n(\varepsilon)$ such that for all $n > n(\varepsilon)$

$$\sum_{k \in \mathbf{A}_n} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) + \sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) < \varepsilon.$$

Consider an arbitrary $\varepsilon > 0$. Let us divide the set \mathbf{A}_n into subsets $\Gamma_n(m_1)$ and $\Delta_n(m_1)$ in the following way:

$$\Gamma_n(m_1) = \{k \in \mathbf{A}_n\} : \left| \frac{k}{n} - m_1 \right| < \tilde{\delta}(m_1, m_2),$$

$$\Delta_n(m_1) = \{k \in \mathbf{A}_n\} : \left| \frac{k}{n} - m_1 \right| \geq \tilde{\delta}(m_1, m_2),$$

where $\tilde{\delta}(m_1, m_2)$ is a certain positive function of m_1 and m_2 defined further. Our aim is to prove that there exists such $n(\varepsilon)$ that for all $n > n(\varepsilon)$:

$$\sum_{k \in \Delta_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{4}, \quad (10)$$

$$\sum_{k \in \Gamma_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{4}, \quad (11)$$

$$\sum_{k \in \overline{\mathbf{A}_n}} B_{n,k}(m_1) < \frac{\varepsilon}{2}. \quad (12)$$

Firstly, let us consider inequality (10). We had already mentioned that for all $t \in (0, 1) \cup (1, \infty)$ $f(t) \geq 0$, and it is easy to see that for considered values of t $f(t) < 2$. So, using the above mentioned property of Bernstein basis polynomials (9), we have:

$$\sum_{k \in \Delta_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) \leq \frac{1}{2n\tilde{\delta}^2(m_1, m_2)}.$$

Let $n_{1,1}(\varepsilon)$ be sufficiently great. Then for all $n > n_{1,1}(\varepsilon)$ the considering inequality (10) is held.

Secondly, let us consider inequality (11). On the one hand, it is not hard to find such $\delta(\varepsilon)$ that

$$\forall t \in (0, \delta(\varepsilon)) : f(t) = \frac{t^2 \cdot \log\left(\frac{t}{t+1}\right) + \log(t+1)}{1-t} < \frac{\varepsilon}{4}.$$

On the other hand, for sufficiently great values of n for all $k \in \mathbf{\Gamma}_n(m_1)$ we have $t_{n,k}(m_1, m_2) < \delta(\varepsilon)$. It follows that for great values of n we obtain

$$\sum_{k \in \mathbf{\Gamma}_n(m_1)} B_{n,k}(m_1) \cdot f(t_{n,k}(m_1, m_2)) < \frac{\varepsilon}{4} \cdot \sum_{k=0}^n B_{n,k}(m_1) = \frac{\varepsilon}{4}.$$

Let $\delta(\varepsilon) = \min\left(\frac{\sqrt{1+(\varepsilon/2 \cdot \ln(2))^2} - 1}{\varepsilon/2 \cdot \ln(2)}; \frac{1}{2}\right)$. Then for $t \in (0, \delta(\varepsilon))$ we have $t < \frac{\varepsilon \cdot \ln(2)}{4}(1 - t^2)$. As far as $0 < t \leq \frac{1}{2} < 1$, we obtain

$$\frac{t - \frac{t^2}{t+1}}{(1-t) \cdot \ln(2)} = \frac{t}{(1-t^2) \cdot \ln(2)} < \frac{\varepsilon}{4}.$$

If we combine this with Proposition 1, then for all $t \in (0, \delta(\varepsilon))$ we have

$$f(t) = \frac{t^2 \cdot \ln\left(\frac{t}{t+1}\right) + \ln(t+1)}{(1-t) \cdot \ln(2)} < \frac{t + t^2 \cdot \left(-\frac{1}{t+1}\right)}{(1-t) \cdot \ln(2)} = \frac{t - \frac{t^2}{t+1}}{(1-t) \cdot \ln(2)} < \frac{\varepsilon}{4}.$$

Now we need to find such $n_{1,2}(\varepsilon)$ that for all $k \in \mathbf{\Gamma}_n(m_1) : t_{n,k}(m_1, m_2) < \delta(\varepsilon)$. By definition,

$$t_{n,k}(m_1, m_2) = \left(\frac{m_2}{m_1}\right)^k \cdot \left(\frac{1-m_2}{1-m_1}\right)^{n-k}.$$

As far as $\frac{m_2}{m_1} > 1$, $t_{n,k}(m_1, m_2)$ strictly increases with respect to k . So, if $\left|\frac{k}{n} - m_1\right| < \tilde{\delta}(m_1, m_2)$ then

$$t_{n,k}(m_1, m_2) < \left(\left(\frac{m_2}{m_1}\right)^{m_1 + \tilde{\delta}(m_1, m_2)} \cdot \left(\frac{1-m_2}{1-m_1}\right)^{1-m_1 - \tilde{\delta}(m_1, m_2)}\right)^n.$$

Consider the right side of this inequality. Note that

$$\left(\frac{m_2}{m_1}\right)^{m_1 + \tilde{\delta}(m_1, m_2)} \cdot \left(\frac{1-m_2}{1-m_1}\right)^{1-m_1 - \tilde{\delta}(m_1, m_2)}$$

strictly increases with respect to $\tilde{\delta}(m_1, m_2)$. It is equal to 1 when $\tilde{\delta}(m_1, m_2) = \tilde{\delta}^*(m_1, m_2)$, where

$$\tilde{\delta}^*(m_1, m_2) = \left(\frac{\ln((1-m_2)/(1-m_1))}{\ln((1-m_2)/(1-m_1) \cdot m_1/m_2)} - m_1\right).$$

According to Proposition 2,

$$\left(\frac{m_2}{m_1}\right)^{m_1} \cdot \left(\frac{1-m_2}{1-m_1}\right)^{1-m_1} < 1,$$

we see that for $\tilde{\delta}(m_1, m_2) \in (0, \tilde{\delta}^*(m_1, m_2))$ we have

$$\left(\frac{m_2}{m_1}\right)^{m_1 + \tilde{\delta}(m_1, m_2)} \cdot \left(\frac{1 - m_2}{1 - m_1}\right)^{1 - m_1 - \tilde{\delta}(m_1, m_2)} < 1.$$

Let $\tilde{\delta}(m_1, m_2) = \frac{\tilde{\delta}^*(m_1, m_2)}{2}$. Now it is easy to put $n_{1,2}(\varepsilon)$ such that for all $n > n_{1,2}(\varepsilon)$ inequality (11) is held.

Finally, let us find such $n_2(\varepsilon)$ that for $n > n_2(\varepsilon)$ condition (12) is satisfied. We have already seen that $|\mathbf{A}_n| \leq 1$, and the equality is held when

$$k_0 = n \cdot \frac{\ln((1 - m_1)/(1 - m_2))}{\ln(((1 - m_1) \cdot m_2)/((1 - m_2) \cdot m_1))};$$

is an integer. Let us denote the right side as $n \cdot c(m_1, m_2)$ and write k_0 as $k_0 = n \cdot c(m_1, m_2)$.

Referring to the standard way of proving the Stirling's formula, we can write the following inequality:

$$\sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n \leq n! \leq e \cdot \sqrt{n} \cdot \left(\frac{n}{e}\right)^n.$$

It follows that

$$\binom{n}{k} \leq \frac{e}{2\pi} \sqrt{\frac{n}{k(n-k)}} \cdot \left(\frac{n}{k}\right)^k \cdot \left(\frac{n}{n-k}\right)^{n-k}.$$

So, it is now easy to see that

$$B_{n,k}(m_1) \leq \frac{e}{2\pi} \sqrt{\frac{n}{k(n-k)}} \cdot \left(\frac{n \cdot m_1}{k}\right)^k \cdot \left(\frac{n \cdot (1 - m_1)}{n - k}\right)^{n-k}.$$

Substituting $k_0 = n \cdot c(m_1, m_2)$ for k in the last inequality, we get

$$B_{n,k_0}(m_1) \leq \frac{e}{2\pi} \sqrt{\frac{1}{n \cdot c(m_1, m_2)(1 - c(m_1, m_2))}} \cdot \left(\left(\frac{m_1}{c(m_1, m_2)}\right)^{c(m_1, m_2)} \cdot \left(\frac{1 - m_1}{1 - c(m_1, m_2)}\right)^{1 - c(m_1, m_2)}\right)^n. \quad (13)$$

It follows from Proposition 2 that

$$\left(\frac{m_1}{c(m_1, m_2)}\right)^{c(m_1, m_2)} \cdot \left(\frac{1 - m_1}{1 - c(m_1, m_2)}\right)^{1 - c(m_1, m_2)} < 1.$$

Now using (13) it is not hard to put $n_2(\varepsilon)$ such great that for $n > n_2(\varepsilon)$ inequality (12) is held.

Finally, let $n(\varepsilon) = \max\{n_{1,1}(\varepsilon), n_{1,2}(\varepsilon), n_2(\varepsilon)\}$. Now for all $n > n(\varepsilon)$

$$\max_{\mathbf{T} \in \mathcal{M}, m \in \mathbb{N}} \{I(\mathbf{X}^{(\mathbf{m})}; \Theta)\} - I(\mathbf{X}^{(\mathbf{n})}; \Theta) < \varepsilon.$$

The theorem is proved. □

5 Conclusions

In the paper the problem of obtaining classical information about the pure qubit state using a single qubit binary test has been considered. It has been demonstrated that the maximum of information is reached if and only if the using measurement is projective. The maximum value of information has been calculated:

$$\max_{\mathbf{T} \in \mathcal{M}, n \in \mathbb{N}} \left\{ I(\mathbf{X}^{(n)}; \Theta) \right\} = \frac{2 \ln 2 - 1}{2 \ln 2}.$$

It follows, in particular, that to distinguish two arbitrary pure qubit states using a single binary test it is necessary to have at least four pairs of qubits prepared in the considered states.

It has been shown that the maximum of reachable information can be attained asymptotically using an arbitrary consecutive qubit binary test. Thus, if we have a certain measuring instrument performing a qubit binary test, we can obtain an amount of information arbitrary close to the maximum.

As known [3, 6], Yu. Manin and R. Feynman proposed to use quantum systems to simulate others quantum systems. The results obtained in the paper show that this idea should be refined: one should take into account all dependences between an input data, a behaviour of a simulating system, and a structure of an output data.

Our further research will deal with generalizing results of the paper for the case of an n -level quantum system and a measurement with m outcomes.

References

1. Cover, T.M., Thomas, J.A.: Elements of Information Theory. John Wiley & Sons, Inc. (1991)
2. Davies, E.B.: Information and Quantum Measurement. IEEE Trans. Inf. Theory IT-24, 596–599 (1978)
3. Feynman, R.P.: Simulating Physics with Computer. Int. J. Theor. Phys., 21, 467–488 (1982)
4. Holevo, A.S.: Bounds for the Quantity of Information Transmitted by a Quantum Communication Channel (in Russian). Problemy Peredachi Informatsii, vol. 9, 3, 3–11 (1973)
5. Holevo, A.S.: Statistical Structure of Quantum Theory. Springer-Verlag, Berlin (2001)
6. Manin, Yu.I.: Mathematics as metaphor: selected essays of Yuri I. Manin. AMS (2007)
7. Natanson, I.P.: Constructive function theory. Vol. 1, Ungar, New York (1964)
8. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information, 10th Anniversary edition. Cambridge University Press, Cambridge (2010)
9. Shannon C.E., Weaver W.: The Mathematical Theory of Communication. University of Illinois Press, Urbana (1949)
10. Thawi M., Zholtkevych G.M.: About One Model of Consecutive Qubit Binary Testing. Bull. Khark. Nat. Univ., vol. 890, 13, 71–81 (2010)

A Data Transfer Model of Computer-Aided Vehicle Traffic Coordination System for the Rail Transport in Ukraine

Denis B. Arkatov¹

¹National Technical University "Kharkov Polytechnic Institute", Kharkov, Ukraine

denarkatov@gmail.com

Abstract. This paper gives a general layout of subsystem operation used for the rolling stock traffic coordination. A principle of selection of information technologies applied for the realization of the communication and data transfer system has been described. GSM and GPRS technologies that are used for the transmission of navigation information on a vehicle locus and for the information exchange in the system have been described. To provide a required level of transmission capacity through the frequency reuse the guard period between the base transceiver stations has been calculated. To calculate the GPRS channel capacity and that of information exchange via the Internet GPRS network an algorithm for the simulation modeling of packet arrival in the prescribed time space has been constructed. Using this algorithm an experiment was carried out for different intensity values of packet arrival to the system.

Keywords. Traffic coordination, information technologies, transmission capacity, algorithm, simulation model, communication channel

Key terms. Development, Model, MathematicalModel

1 Introduction

The traffic control automation is a topical and up-to-date problem of the rail transport in Ukraine. An important part of this problem is the development of algorithms for the coordination and control of the large amount of rolling stock, situated in a zone of railway traffic control points. An important requirement set to such algorithms is to provide safe and regular traffic of the whole collection of trains and to make optimal decisions from the economic standpoint.

To provide safe traffic for rolling stocks at the stage of departure or arrival a certain (permissible minimum) time interval should be provided, which is always taken into consideration while making a train timetable. Despite this fact the train timetable is sometimes disturbed due to many reasons and situations arise when the rolling stocks are found to be undivided by the safe time interval and the groups of conflict-

ing trains are formed. In the case of high traffic intensity the groups of conflicting trains at stations or at a train overtaking locus can be rather large. On the other part the malfunction of train timetable can result in subsequent traffic disturbance (train delay, train overtaking conflicts, etc.) The above problems can be resolved through the automation of operative traffic control of rail transport. The experience gained by foreign countries [1,2] shows that the efficient solution of traffic control with regard to the rail transport is only possible if the latest information technologies are used.

The use of inexhaustible potential of railway information systems for the benefit of entire transport system of the country allows to reduce control costs required for the management and realization of domestic and international traffic. This also provides a considerable improvement in the quality of transport and logistics-related services and safety of traffic.

The examples of computer-aided systems that can fully or partially solve the above problems are the European Train Control System (ETCS) [3], American Positive Train Control (PTC) [4] and ILSD-U system [5] used by Russia.

A specific feature of the developed computer-aided system used for the rolling stock (RS) traffic coordination is the introduction of contemporary satellite technologies, communication and data transfer systems into a routine work of rail transport in Ukraine. The information exchange technology can be described as follows.

A GPS/GPRS –modem receives the navigation information from satellites, after that using the GPRS technology the coordinates, current speed and other information are transmitted via the operator's server of mobile communication to the database server. The RS locus data are transmitted to the workstation of railway station dispatcher and to the workstation of railway dispatcher.

The data should be processed in a real time mode. An important aspect is that a time interval during which these data gain currency should be taken into consideration. This fact considerably constrains not only the algorithm used for the solution of coordination problem but also data transfer technologies.

This paper consists of several sections. First of all, the technologies used for the data collection, processing and transfer will be described. Then, we will estimate the channel data transfer capacity and do appropriate computations that prove the obtained results.

2 Data Transfer, Processing, and Collection Technology

At the present time the railway adopts the computer-aided system (CAS), which includes the following basic subsystems: onboard intellectual system, which provides positioning, control and information support for the rolling stock; surface intellectual system (SIS), which provides control and coordination in a real time mode; communication and data transfer system based on the mobile GSM communication; navigation charts that reflect a real railway infrastructure. The SIS structure and CAS system on the whole are given in Fig.1.

A special place in SIS is occupied by the information system designed for the rolling stock traffic coordination, which is the subject of this research. The navigation

data collection system is used for the automatic identification of a rolling stock locus and also for securing the safety of traffic. The information about a rolling stock locus is required for the optimal use of traffic and carrying capacity of railways and for the elimination of dangerous situations (dangerous passing approach, passing the traffic lights with forbidden signals, siding motion, exceeding the allowable speed in the places of its restriction, etc.) A locus of each RS is transmitted to the traffic control service via the navigation system for further data processing and traffic control. The basic principles of monitoring are the safety of traffic, time fulfillment of a transportation schedule and costs minimization.

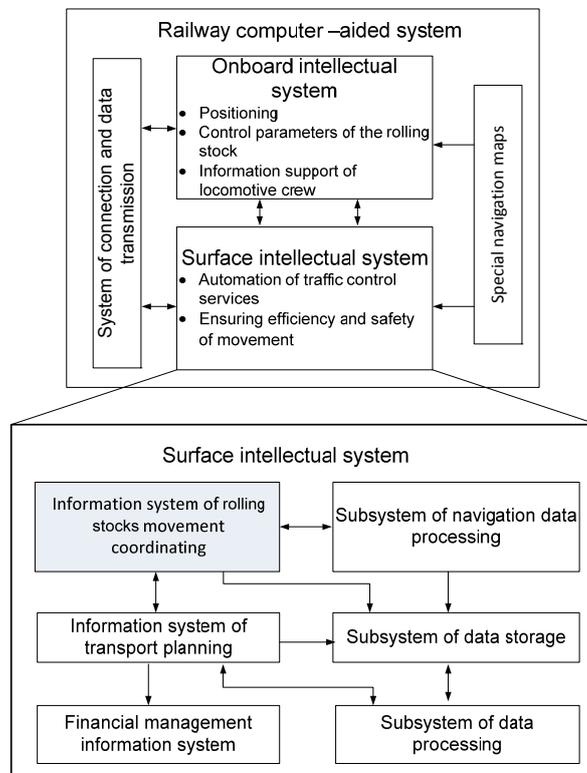


Fig. 1. CAS and SIS structures

In order to create the appropriate computer-aided communication and data transfer system for the rail transport the adoption of communication standard is required to meet the entire system operation requirements, in addition to the use of satellite navigation. The required communication system should provide high safety, reliability and solutions proved in practice and it should also be an innovative and high performance system. The costs required for the total system deployment should be reduced to a minimum.

Taking into consideration the European experience gained in the realization of similar data transfer systems telecommunication technologies should be developed meeting the GSM-R standard (Global System for Mobile Communications-railway)[6,7], which engineering and economic efficiency has been proved not only by tests but also by real application for different railway systems in developed European countries. However, at this stage of the development of information technologies in Ukraine we can make to the conclusion that the realization of formulated problem is possible and it is highly recommended to use the GSM mobile communication standard. The GPRS technology is recommended for the data transfer.

The GPRS networks split the transmitted information into individual packets that are delivered from a transmitter to a receiver. If errors have been detected the received packets can be transmitted once again. The original message is designed by the receiver party using the obtained packets. The data transfer in packet-switching networks differs from the data transfer in channel-switching networks in the way that the required channel resource is allocated exclusively for the time of transmission of appropriate information packets. The rest of the time it is at disposal of a network. In the case of GSM/GPRS networks this allows us to use one physical channel for the transmission of packets to several subscribers and to simultaneously allocate several physical channels for the transmission of packets to one subscriber. The packets are transmitted aside from each other in different directions.

The GPRS defines the effective use of a channel resource. The same physical channel is provided for the group of subscribers to receive messages sent from the base transceiver station (BTS) to a mobile station (MS) and packets are transmitted as soon as they arrive depending on the volume of information and the priority of subscribers. Each packet contains an identifier or address, which is used for the delivery. A subscriber is continuously connected to the packet network, which provides for him a virtual channel. It becomes a real (physical) radio channel during the packet transmission. The rest of the time this physical channel is used for the transmission of packets of other users.

Due to the fact that the same channel resource is used by several subscribers and during the communication session the packets of different users can simultaneously arrive the waiting list of transmitted packets can be originated, which will result in the communication delay. The allowable value of packets delay is one of the attributes defining the quality of a subscriber service.

The paper [8] defines three classes of delay depending on the delay norms and packet length. The top priority is assigned to the Class 1, the normal priority is given to the class 2 and the class 3 enjoys the least priority. The delays have not been defined for the Class 4, because the service of packets of this class is performed adhering to the "best effort" principle.

The intensity values of packets arrival can be selected on the basis of statistical research. A statistics shows that a number of packets arriving per time units to the input of GPRS circuit changer can vary in a wide range from hundreds of packets /s at input sites to several thousands of packets/s at backbone sites.

To provide the appropriate level of data throughput we use the basic principle of the construction of cellular communication networks, which is based on the frequency

reuse [8]. The main essence of it consists in that the neighboring (adjacent) cells of a mobile communication system use different frequency systems and in non-adjacent cells located at sufficient distance from each other the used frequency bands are repeated. In practice the cities and regions with a solid cellular coating use clusters in which each cell is divided into the three sectors, using the directional radiation antenna with the directional pattern width of 120° .

The base stations that allow frequency reuse are located at a distance D from each other. This D distance is measured between the centers of hexagonal cells and it is called a guard interval.

Proceeding from geometrical reasons the parameter D can be defined as follows:

$$D = R\sqrt{3\eta},$$

where R is a radius of circle circumscribed around the regular hexagon; η is a coefficient of the frequency reuse. The $\frac{D}{R}$ ratio is defined as a reduction factor of channel noises.

Thus, for the optimal frequency reuse and an increase in the GPRS channel capacity in Ukraine the guard interval should be:

$$D = 15\sqrt{3} \cdot 3 = 45 \text{ (km)}$$

3 Case-Study for Capacity Evaluation

The developed computer-aided system used for the rolling stock traffic coordination should take into consideration, while transmitting the navigation information to the mobile operator server, not only the capacity of GPRS channels but also that of cable or fiber-optic channel of the Internet network, which is used for the transmission of data to the database server.

By analogy to the mobile communication it is necessary to consider a problem related to the average duration of delays in the packet switch.

The term "packet switch" implies here a concentrator (statistical multiplexor), a virtual packet switch (network X*25, Frame Relay, and ATM network) and a router (IP network). The packet switch can be represented as an element with many input and output channels (switch/router). Using the Kendall notation such network elements can be presented by queuing systems of G/G/1 or G/G/n type (arbitrary probabilistic distributions describing the incoming stream of customers (in our case packets or protocol blocks) and the time of their serving time. (Let us note that models with one server, i.e. G/G/1 are often used for the analysis of packet switches).

Let's assume that in the general case applications arrive to the system input in compliance with the Poisson distribution law, whose service time is an arbitrary value. Then the average queue length in the system with the infinite buffer size (M/G/1) is calculated using the classic Khintchine-Pollaczek formula [9]:

$$\bar{t}_q = \frac{\bar{q}}{\lambda} = \bar{t}_s \times \left[1 + \rho \frac{1 + C_s^2}{2(1 - \rho)} \right] \tag{1}$$

where \bar{q} is an average queue length in the considered system (including protocol blocks PB);

$\rho = \frac{\lambda}{\mu}$ is the M/G/1 system load intensity, $\rho < 1$;

λ, μ are the intensity values of the PB arrival and service in the system, accordingly;

\bar{t}_s is an average time of PB service in the system;

$C_s^2 = \frac{D(t_s)}{(\bar{t}_s)^2}$ is a quadratic coefficient of service time variation equal to the ratio of service time variance and squared expectation value.

The values required for computations using formula (1) were obtained through the simulation modeling of data transfer in the Internet network, which algorithm is given in [1].

A one-channel queuing system (QS) of the M/G/1 type has been taken as an example. The arrival of the Poisson stream of applications with the constant service time τ_j has been simulated. The numerical experiment, which was carried out, allows us to come to the conclusion that there is no loss of data packets.

The constructed graphs show the time of the arrival of customers in a queue and the time of channel teardown from the previous application at the application arrival intensities of $\lambda = \frac{1}{10}$ (see Fig.2.a) and $\lambda = \frac{1}{20}$ (see Fig.2.b)

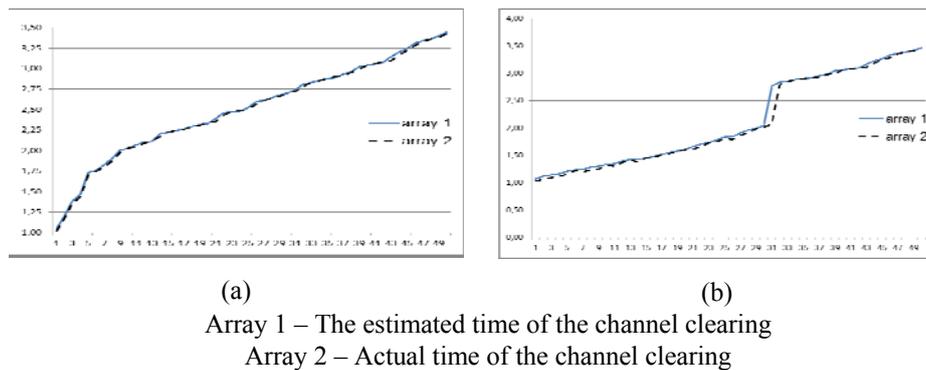


Fig. 2. QS simulation with an intensity of $\lambda = \frac{1}{10}$ and $\lambda = \frac{1}{20}$

The given Figures show that an increase in intensity of arriving customers does not result in the lack of channel capacity. This proves the correct choice of information

technologies for the realization of computer-aided system used for the rolling stock traffic coordination.

To define the data transfer channel capacity in the Internet network the network duration and delay should be calculated.

Due to the fact that the data transfer channel-switching center services packets it can be simulated using the system with a constant service time of a M/D/1 type.

$$\bar{t}_q = \bar{t}_s \times \left(1 + \frac{\rho}{2(1-\rho)} \right).$$

$$\text{At } \lambda = \frac{1}{10} \text{ and } \mu = \frac{1}{5}$$

$$\rho = \frac{\lambda}{\mu} = \frac{1}{2}$$

Let t in the given example be equal to 0,05. As a result we get:

$$\bar{t}_q = 0,05 \times \left(1 + \frac{1}{2} \right) = 0,075,$$

which corresponds to the first service class.

$$\text{At } \lambda = \frac{1}{20} \text{ and } \mu = \frac{1}{15}$$

$$\rho = \frac{\lambda}{\mu} = \frac{3}{4}$$

Let t in the given example be equal to 0,05. As a result we get:

$$\bar{t}_q = 0,05 \times \left(1 + \frac{3}{2} \right) = 0,125,$$

which corresponds to the first service class.

4 Conclusions

Today the railways are the major branch of the economy of Ukraine and they serve as a basis of the Ukrainian transportation system. Due to the rapidly changing demands for freight services and carriage of passengers a permanent control over the required amount of rolling stocks should be investigated. The proper amount of such vehicles can be defined through the traffic analysis.

The information on the timetable of railway traffic and actual amount of rolling stocks involved in traffic serves as a source information for further computations of the capacity of a communication and data transfer system of the computer-aided system. This will provide the fulfillment of actuality condition of obtained data and the solution of the traffic coordination problem in a real time mode.

To provide the functioning of computer-aided system on the whole the following algorithms should be realized. The algorithm used for the determination of the amount of conflicting trains is the first step towards the problem solution. It allows for the detection of those rolling stocks for which conflict-free conditions are not observed due to different reasons (timetable violation, technical malfunction, etc.). A decomposition algorithm is required for the reduction of dimension of a solved problem that would allow the reduction of time interval of data acquisition. The obtained data are used for the generation of control action and for the reduction of loading for the used transmission channel, providing thus the observance of principles of control in a real time mode. An algorithm of the problem related to the coordination of rolling stock traffic allows for the elaboration of such control actions that provide not only the elimination of a definite conflict but also the fulfillment of conflict-free conditions for the definite rolling stock in the future. An algorithm of data transfer in a real time mode is the basis of information exchange between the rolling stock and the database server. Our further research is targeted at the development of algorithmic support of communication and data transfer system to solve the problem on the coordination of traffic of rolling stocks.

This paper gives the description of GSM and GPRS technologies that are used for the transmission of navigation information on the locus of rolling stocks and also for the information exchange in the system. It has been noted that in order to provide the required capacity level we applied the main principle used for the construction of cellular communication networks, i.e. the frequency reuse. The cluster structure with a template of 3/9 has been constructed and the guard interval between the BSs used for the mobile communication in Ukraine has been calculated.

To calculate the capacity of GPRS channels and that of information exchange channel in the GPRS Internet network the algorithm of simulation modeling of arriving packets during the prescribed time interval has been constructed. On the basis of this algorithm we carried out an experiment for different intensity values of packets arrival to the system. Using the analysis data we can come to the conclusion that an increase in the intensity of arriving customers causes no lack of channel capacity.

Our further research is targeted at the development of an algorithm for the data processing in a real time mode, and also at the development and testing of the information system designed for the rolling stock traffic coordination. We will also delve into the computation of efficiency estimates to evaluate the introduction of information technologies into the rail transport operation.

References

1. Arkatov, D. B.: Models and methods for automation of dispatching management for railways of Ukraine. Modeli i metodi avtomatizacii dispetcherskogo upravleniya dlya zheleznodorozhnogo transporta Ukraini, Vostochno-Evropeyskiy zhurnal peredovikh tekhnologiy, Eastern European journal of Enterprise Technologies, N 1/10 (61), pp. 61–63 (2013)
2. Arkatov, D. B.: Synthesis models of coordination of movement of mobile railway transport of Ukraine. Sintez modeley koordinacii dvizheniya podvizhnikh sredstv

zheleznodorozhnogo transporta Ukraini, Vostochno-Evropeyskiy zhurnal peredovikh tekhnologiy, Eastern European journal of Enterprise Technologies, N 4/3 (58), pp. 58–60 (2012)

3. ETCS requirements specification and validation: the methodology, http://www.era.europa.eu/Document-Register/Documents/ETCS_methodology_v_1_2.pdf (accessed 20 February 2013)
4. Positive train control (2013), http://en.wikipedia.org/wiki/Positive_train_control (accessed 20 February 2013)
5. Integrated locomotive safety device – unified, <http://www.irz.ru/products/20/70.htm> (2012) (accessed 20 February 2013)
6. GSM-R (2013), <http://en.wikipedia.org/wiki/GSM-R> (accessed 20 February 2013)
7. GSM technical specification, <http://www.tfn.net/techno/smartcards/gsm11-11.pdf> (1995) (accessed 20 February 2013)
8. QoS in GPRS, <http://doc.utwente.nl/18117/1/00000039.pdf> (1999) (accessed 20 February 2013)
9. Tijms, H. C. A first course in stochastic models / Henk C., Tijms.p. cm. Includes bibliographical references and index.

Quantitative Estimation of Competency as a Fuzzy Set

Leonid Vasylevych¹ and Ivan Iurty¹

¹Borys Grinchenko Kyiv University,
Department of Information Technology and Mathematical Sciences

lvasilevich@mail.ru, iurty@ukr.net

Abstract. The authors of this paper have used the assessment of competence as a fuzzy discrete set consisting of essential capacities. There has been proposed a procedure of competence quantitative estimation on the basis of discrimination index of discrete fuzzy sets fixed on one totality. A linguistic variable “Competency coefficient” has been used here for making appropriate decisions on the grounds of competency quantitative estimation. Assessment of a person’s competency is proposed as a fuzzy discrete set consisting of necessary abilities as its values. Using such competency assessment allows to estimate persons’ competency quantitatively and to compare them.

Keywords. Competency-oriented education, competence, capacities, fuzzy discrete set, linguistic variable, membership functions, fuzzification, scalar capacity of any fuzzy discrete set

Key terms. MathematicalModelling, MathematicalModel, FormalMethod

1 Introduction

The analysis of world education development tendencies demonstrates [1,3] competency-oriented education trend increase. Moreover, competency, which is not only defined by knowledge, abilities, skills but also by considerably greater quantity of factors (coefficients), becomes a major category both in education system and in the job-market. Competency also includes the ability to obtain, to analyze and to revise information; to learn through one’s lifetime; to change in compliance with the job-market demands [1].

Thus, the quantitative estimation of competency necessary for making appropriate decisions is a multicriterial problem, and therefore we need here to derive an integral estimation of competency. Since there is no methodology of working out this problem, it makes the article topical for in it an integral index of a person’s competency coefficient is estimated on the basis of the new competency assessment as a fuzzy discrete set of which essential capacities are values.

Published works analysis. In the work [1], the key competencies concept has been considered and three key competencies have been analyzed (specified by the Organization of Economic Cooperation and Development (OECD) representatives), which are: autonomous activity; interactive facility use; ability to work in socially hetero-

genic groups. Federal Statistics Department of Switzerland and National Center of Education Statistics of the USA and Canada within the program named Definition and Selection of Competencies-- Theoretical and Conceptual principles (“DeSeCo”)) summarized respective scientific results and different countries’ practices. In the work [3] we give a review of works on the topic. But in all those works the qualitative approach to the named subject is solely used, but methods of quantitative competency estimation have never been given.

Thus, the aim of this work is to develop methods of quantitative estimation of competency on the grounds of its assessment as a fuzzy discrete set [2] consisting of essential capacities.

Main results. Competency is defined in UNESCO publications as a combination of knowledge, abilities, values and attitudes used in everyday life. Therefore qualitative assessment of a person’s competency means his (her) ability to perform professional duties or some functions efficiently. But this definition does not give a possibility to estimate expert’s competency on quantitative basis. That is why we proposed to use the following person’s competency definition.

Definition 1. A person’s competency is a finite discrete fuzzy set consisting of abilities necessary for a job position or functions necessary for a respective position. Membership functions of the set elements characterize the level of this competency innateness to the person.

Definition 2. Abilities are necessary features, characteristics, faculties, qualities, knowledge, techniques, skills and other traits which a person needs to perform duties or functions at a respective position efficiently.

Thereby, in the beginning, we need to define at the discrete set of abilities $Y = \{y_j : j = \overline{1, m}\}$ membership functions $\mu_D(y_i) \in [0; 1]$ of the fuzzy set D “Requirements necessary to perform duties or functions at a respective position efficiently”. These membership functions characterize credibility, priority and importance of a respective ability for a respective position or function.

Further we will use the notation of the discrete fuzzy set D in the form [2]:

Table 3. Designation D discrete fuzzy set

D=	y_i	y_1	y_2	y_3	...	y_n
	$\mu_D(y_i)$	$\mu_D(y_1)$	$\mu_D(y_2)$	$\mu_D(y_3)$...	$\mu_D(y_n)$

or $D = \langle (y_1 / \mu_D(y_1)); (y_2 / \mu_D(y_2)); (y_3 / \mu_D(y_3)); \dots (y_n / \mu_D(y_n)) \rangle$.

The set of abilities and respective membership functions will be different for each position. When we specify the Y set we need to apply the Pareto principle, which points that 20% of factors define 80% of the result. In practice, implementation of this principle will lead to the effect that abilities with membership functions less than 0.5 will not be included into the D set. The task of specifying the set of abilities and respective membership functions refers to the task of knowledge estimation by experts and demands creation of respective questionnaires.

As an example, let us specify an IT teacher’s information technology competency in the form of a fuzzy set D:

Table 4. Example D representation of discrete fuzzy set

D=	Y_i	y_1	y_2	y_3	y_4
	$\mu_D(y_i)$	1	0.9	0.7	0.8

in which y_1 - ability to work in Word environment; y_2 is the technique of work in Excel environment; y_3 is the technique of work in Excess environment; y_4 is special software skills (e.g. working out optimization tasks).

To perform a quantitative estimation of a particular teacher’s competency it is necessary to estimate his abilities y_i . To do so, tests, interviews, exams, respective lessons control and other means can be recommended. Competency grades (their membership functions estimation) can be shown on the scale from 0 to 1. This process is called fuzzification. Hereby, for each person (teacher), we can define in the form of a fuzzy set his personal fuzzy vector of abilities, which defines his competency. To define $\mu_A(y_i)$ a group of experts can be used who, after analyzing the person, answer the question: “Is ability y_i attributable to the person?” If the LD expert of L experts give a positive answer, then

$$\mu_A(y_i) = \frac{L_D}{L}. \tag{2}$$

As a rule this question does not have a single-value answer, so, experts can use both binary logic ($\mu_{A\gamma}(y_i)$ is either 0 or 1, where γ is an expert’s number) and fuzzy logic (multiple-valued verity scale). In so doing they index the value of $\mu_{A\gamma}(y_i) \in [0; 1]$ (subjective estimate). If quantity of the experts is L, then in the capacity of $\mu_A(y_i)$ we accept weighted arithmetic mean value of these estimates:

$$\mu_A(y_i) = \frac{\sum_{\gamma=1}^L k_{\gamma} \mu_{A\gamma}(y_i)}{\sum_{\gamma=1}^L k_{\gamma}}, \tag{3}$$

where k_{γ} is the γ expert’s competency estimate.

For quantitative comparison of different persons’ competencies we need, firstly, to compare in pairs finite discrete fuzzy sets D “Demands necessary to perform duties or functions at a particular position efficiently” and A_j “the j person’s competency” which are specified at one totality Y.

To compare these finite discrete fuzzy sets in pairs it is possible to use the estimate $P(D, A_j)$ of difference between D and A_j , which is reduced to the estimate of the traverse of $\overline{D} \cap A_j$ or $D \cap \overline{A_j}$ [2]:

$$P(D, A_j) = \frac{|\overline{D} \cup A_j| - |\overline{D}|}{|D|}, \tag{4}$$

where the $|\dots|$ sign means scalar capacity of any fuzzy discrete set B [2]:

$$|B| = \sum_{x \in X} \mu_B(x) \tag{5}$$

operation \overline{B} of complementing the fuzzy set B is defined by the membership function [2]

$$\mu_{\overline{B}}(y) = 1 - \mu_B(y), \quad \forall y \in Y \tag{6}$$

operation of two fuzzy sets unification ($C = B \cup K$) has the membership function [2]

$$\mu_C(y) = \max(\mu_B(y); \mu_K(y)), \quad \forall y \in Y. \tag{7}$$

In so doing $P(D, A_j)$ as a rule is not equal to $P(A_j, D)$. This attribute is used to compare fuzzy sets specified at one totality: if $P(D, A_j) > P(A_j, D)$, then the fuzzy set $D < A_j$ and vice-versa.

A person's abilities, which have some membership functions' value greater than the value of respective abilities' membership functions in the D set, must not compensate small values of the A_j set membership functions. To avoid this it is necessary to perform the A_j set normalization: membership functions values of the A_j set which exceed respective values in the D set have to be equated to respective membership functions' values of the D set. Thereby it is necessary to insert the normalized fuzzy set A_{jn} into the (3) formula.

Let us perform a comparison of two persons' competencies. Let us define one person's competency by means of a fuzzy set $A_1 = \langle (y_1/0.6); (y_2/0.9); (y_3/0.7); (y_4/0.9) \rangle$ and the other person's competency by means of a fuzzy set $A_2 = \langle (y_1/0.8); (y_2/1); (y_3/0.5); (y_4/0.9) \rangle$.

After normalization we have: $A_{1n} = A_1 = \langle (y_1/0.6); (y_2/0.9); (y_3/0.7); (y_4/0.9) \rangle$;

$$A_{2n} = \langle (y_1/0.8); (y_2/0.9); (y_3/0.5); (y_4/0.8) \rangle.$$

The estimate of the difference $P(D, A_1)$ is equal to (3):

$$P(D, A_1) = \frac{0.6 + 0.9 + 0.7 + 0.9 - 0.6}{1 + 0.9 + 0.7 + 0.9} \approx 0.735.$$

The estimate of the difference $P(A_1, D)$ is equal to:

$$P(A_1, D) = \frac{1 + 0.9 + 0.7 + 0.8 - 0.9}{0.6 + 0.9 + 0.7 + 0.9} \approx 0.806.$$

We propose to calculate competency coefficient K is as the normalized estimate of differences:

$$K = \frac{\min(P(A, D); P(D, A))}{P(A, D)} \tag{8}$$

This coefficient always belongs to [0;1] interval. If $P(A,D) > P(D,A)$ then $K < 1$, and if $P(A,D) < P(D,A)$ then $K = 1$.

After inserting computed estimates into the (7) formula we have:

$$K = \frac{\min(0.806; 0.735)}{0.806} \approx 0.912 .$$

Calculation of competence coefficient K for the second person will give values described below:

$$P(D, A_2) = \frac{3.2 - 0.6}{3.4} \approx 0.765; \quad P(A_2, D) = \frac{3.4 - 0.8}{3.2} \approx 0.813;$$

$$K = \frac{\min(P(A, D); P(D, A))}{P(A, D)} = \frac{\min(0.813; 0.765)}{0.813} \approx 0.941.$$

Thereby, we can conclude that the second person's competency is greater than the first one's.

To define a person's competency level basing on the competency coefficient value it is necessary to specify a linguistic variable (LV) [2] "A person's competency coefficient", which we will determine by means of a tuple $\langle E, E_j, j = \overline{1,5}; \mu_{E_j}(x) \in [0;1]; x = K \in [0;1]; \rangle$.

Terms of "Competency" LV can be: E_1 – very low competency; E_2 – low competency; E_3 – medium competency; E_4 – high competency; E_5 – very high competency. Trapezoidal membership functions of terms can be defined by experts by means of four numbers $\langle a : b : c : d \rangle$, which define each term.

Using trapezoidal membership functions of terms and considering Harrington's scale it is possible to specify "Competency" LV as follows: $E_1 = \langle 0 : 0 : 0.1 : 0.2 \rangle$; $E_2 = \langle 0.1 : 0.2 : 0.3 : 0.4 \rangle$; $E_3 = \langle 0.3 : 0.4 : 0.6 : 0.7 \rangle$; $E_4 = \langle 0.6 : 0.7 : 0.8 : 0.9 \rangle$; $E_5 = \langle 0.8 : 0.9 : 1 : 1 \rangle$.

Let estimate E_j of a term by an γ expert amounts $\hat{E}_{j\gamma} = \langle a_{j\gamma}; b_{j\gamma}; c_{j\gamma}; d_{j\gamma} \rangle$, then in the capacity of membership function E_j of the term we accept a fuzzy quantity

$$E_j = \left\langle \frac{1}{L} \sum_{\gamma=1}^L a_{j\gamma}; \frac{1}{L} \sum_{\gamma=1}^L b_{j\gamma}; \frac{1}{L} \sum_{\gamma=1}^L c_{j\gamma}; \frac{1}{L} \sum_{\gamma=1}^L d_{j\gamma} \right\rangle. \tag{9}$$

To specify terms more appropriately Delphi technique can be applied.

Specifying membership functions lateral branches by straight line segments does not reduce persons' competency estimate's generality but simplifies mathematical operations over fuzzy quantities considerably [4]. In so doing the left $\mu_l(x)$ and the right $\mu_r(x)$ lateral branches of the membership linear function have analytical form respectively:

$$\mu_l(x) = \frac{x - a}{b - a}; \quad x \in [a; b], \tag{10}$$

$$\mu_r(x) = \frac{d-x}{d-c}; \quad x \in [c; d]. \quad (11)$$

For the just made example, we have ascertained that the competence coefficient $K_1=x=0.912$ belongs to E_5 term (very high competency) with membership function (verity) one, and the competence coefficient $K_2=x=0.88$ belongs to E_4 term (high competence with membership function 0.2 and to E_5 term with membership function 0.8).

Algorithm of a person's competency estimation consists of six stages: the preparatory (1 to 4) and operational (5, 6) ones.

1. Specifying the set of abilities $Y = \{y_j : j = \overline{1, m}\}$ for a position or functions.
2. Abilities' membership functions assessment (Specifying D fuzzy set) ((1) and (2) formulae are applied).
3. Ai fuzzy set assessment – “A person's competency”.
4. “A person's competency” LV assessment: $\langle E, E_j, j = \overline{1, 5}, \mu_{E_j}(x) \in [0; 1], x \in [0; 1] \rangle$.
5. A person's competency coefficient computing ((3); (4); (5); (6) and (7) formulae are applied).

Computing a person's competency coefficient's membership functions to respective terms LV “A person's competency” (formulae (9) and (10)).

At the preparatory stage experts are used, who define the notion “Competency” as a discrete fuzzy set, the values of which are abilities necessary for a particular position or functions.

Point 3 demands creating respective techniques, tests, problems and tasks that allow estimating various abilities of a person (to find membership functions of each ability).

At the stage of receiving a person's competency quantitative estimate points 5 and 6 are performed.

To specify A fuzzy set of an expert's antecedent characteristics the expert's questionnaire data, his (her) tests, interviews can be used.

The examined competency estimation methodology based upon using fuzzy sets and a linguistic variable allows resolving several problems: conversion from current qualitative competency assessments to quantitative estimation; multicriteriality of competency estimation problem; impossibility of quantitative measuring certain particular indexes of competency; impossibility of real experiments to estimate different persons' competency.

2 Conclusions

1. Assessment of a person's competency is proposed as a fuzzy discrete set consisting of necessary abilities as its values. Using such competency assessment allows to estimate persons' competency quantitatively and to compare them.
2. A methodology of a person's competency quantitative estimation is proposed.

3. It is proposed to estimate quantitatively persons' competency on the basis of difference coefficient of finite discrete fuzzy sets D "Demands necessary to perform duties or functions at a particular position efficiently".
4. It is proposed to specify an expert's competency coefficient in the form of linguistic variable "Competency".

References

1. Key Competencies: A Developing Concept in General Compulsory Education. EURYDICE. The Information Network on Education in Europe, p. 224 (2002)
2. Pospelov B.A. (ed.): Fuzzy Sets in Management and Artificial Intelligence Models. Science, Moscow (1986)
3. Sysoyeva S.O.: Education and Personality in Post-Industrial World. Monograph. KSPA, Khmelnytsky (2008)
4. Vasylevych, L.F. Malovik, K.N., Smirnov, S.B.: Quantitative Methods of Making Decisions in Terms of Risk. SNUNEP, Sevastopol (2007)

**1.4 Methodological and Didactical Aspects
of Teaching ICT and Using ICT
in Education**

New Approaches of Teaching ICT to Meet Educational Needs of Net Students Generation

Nataliya Kushnir¹, Anna Manzhula¹ and Nataliya Valko¹

¹ Kherson State University, 27, 40 rokiv Zhovtnya St., 73000 Kherson, Ukraine
kushnir@ksu.ks.ua ilovetrees@mail.ru valko@ksu.ks.ua

Abstract. The paper describes the educational needs of modern students as generation Net representatives, highlights the contradiction between their characteristics and traditional ways of teaching ICT disciplines. The paper reports teaching experience and poll results for three years at Kherson State University, Ukraine. The purpose of the paper is to offer new teaching approaches to cope with a generation gap and way to improve the quality of ICT teaching.

Keywords. Generation gap, Generation Net, teaching approaches, ICT discipline

Key terms. ICTTool, TeachingPattern, TeachingMethodology, Capability

1 Introduction

The UNESCO report on Information Technology in Education [1] informs that Ukraine is on the way of "the rapid advancement (progress) of ICT in education that leads to the constant updating of the educational content and the quality of ICT training". However, there is a great amount of problems. Primary it is connected with the fact that educational institutions and teachers in particular are not ready to the transition to the information society: "increased demands for flexibility, mobility and adaptability to the education management system, educational institutions and teachers in the context of rapid changes make it difficult to maintain and improve the quality of educational services."

2 Related Work

Using our teaching experience we identified the common trend of learning styles among the students. Further research is required to investigate digital competence formation among future teachers. Our attempt to find out an unknown factor that has a significant impact on the pedagogical process and to understand the nature of this phenomenon was based on the considering modern students as the representatives of the new generation.

To date, about 40 books and scores of articles and papers have been written on this generation that report the results of international surveys and other research and describe their characteristics. Their impact on education at all levels has been of major interest to researchers and educators. There are about 10 terms to describe the current generation of students [2]: Millennials (Howe and Strauss), Generation Y or Gen Y (Nader), Echo Boomers (Tapscott), Net Generation (Tapscott), Digital Aboriginals (Tarlow and Tarlow), Digital Natives (Prensky), Nexters (Raines and Filipczak), Dot Com Generation (Stein and Craig).

The representatives of this generation were born in 1982-2003. Today's students and post-graduates are aged from 10 to 30. It means that all teachers and institutions that are involved in the education of the students who have grown up in the world of new digital, mobile and high-tech, digital technologies.

The technology itself has had a profound effect on this new generation, unlike on any previous one. In the classroom, students can chat on Skype or write SMS to their friends, take notes on iPad, surf the Internet and read a book on the ReadBook. This behavior can not be fully appreciated by their teachers: it's considered that electronic instruments and digital devices distract students from the "real" study [2]. Majority of today's teachers are representatives of the previous generations. They are using learning models fitted for the teachers themselves but not for the new generation of students.

It was found that representatives of the new generation inherent a wide range of characteristics that are defined a predisposition for becoming a successful educator. Realizing their significance for the education and considering themselves to be an instrument of world changes, they will be strongly motivated to improve the quality of life of the society. Some research study and develop strategies for a retention them in definite professional sphere including education.

Moreover, the representatives of Net generation have solid moral values connected with a family and our society. They are highly motivated to create an open and tolerant society. It's important for new gens' educators to consolidate and maintain youths' system of values. [15].

The results of the 3rd year students' polling showed the low level of professional awareness (Preschool and Elementary School faculty). Only 50% of students have an intention to become teachers [16]. The lack of professional focus among students sets a hard task for educators to make teaching a valuable and desired profession.

Among the characteristics of generation should be noted that Gen Y workers are usually educationally focused and attribute their success to their educational capabilities. They want to have successful careers. They do not like the dress code, demand ICT equipped workplaces and want have a flexible work schedule.

Ronald A. Berk synthesized pertinent research evidence based on ten national and international surveys: EDUCAUSE [4], College Students' Perceptions of Libraries and Information Resources Survey, Greenberg Millennials Study [5], Education Research Institute (UCLA) [3] American Freshman Survey [11], National Center for Education Statistics [9], Net Generation Survey [8], The Net Generation: A Strategic Investigation [13], Nielsen Net View Audience Measurement Survey [2, 10], Pew Internet and American Life Project [6, 7] и Technological preparedness among enter-

ing freshman [12]. The research results from the surveys and aforementioned books has yielded twenty learner characteristics typical for most Net Geners: technology savvy, relies on search engines for information, interest in multimedia, create Internet content, operate at —twitch speed, learn by inductive discovery, learns by trial and error, multitask on everything, short attention span, communicate visually, crave social face-to face interaction, emotionally open: Embrace diversity and multiculturalism, prefers teamwork and collaboration, strive for lifestyle fit, feels pressure to succeed, constantly seeks feedback, thrives on instant gratification, respond quickly and expect rapid responses in return, prefers typing to handwriting. The research results from the surveys and aforementioned books have yielded twenty learner characteristics typical for the most Net Geners.



Fig. 1. Improvement discipline background

We have identified some teaching approaches that are contradictory to the contemporary students' needs. This gap is especially obvious in teaching computer related disciplines. Complete comprehensive step-by-step instructions and exclusively individual learning are no longer efficient. This context led educators to a revision of present teaching strategies.

3 Setting up the Pedagogical Experiment

The only discipline - "New Information Technologies and Technical Facilities of Education" – that concerns the formation of ICT skills is taught for the future teachers of all specialties at Kherson State University. This year the discipline was renamed in "Information Technology" in most curricula.

The teaching experience of teaching the discipline "New Information Technologies and Technical Facilities of Education" in 2011 (109 students) and in 2012 (112 students) at Faculty of Pre-School and Elementary Education (FPPE) allowed us to identify problems that are mainly related to the mentioned contradictions [17].

Table 1. The contradictions of the present ICT teaching approaches to the generation Net characteristics

ICT teaching approach	Generation Net characteristic	Poll results
ICT teaching "from scratch": disregard (neglect) the actual level of the student's ICT skills. As a result, school ICT discipline assignments are duplicated, lab manuals are detailed and tend to be comprehensive.	Tech savvy	84% of respondents have started to use the computer for learning 7 years ago or earlier
Teaching materials are not interactive and update	Relying on search engines for information	About 26% of the students classified (attributed, mentioned, placed, noted) search engines to (as) the most frequently used sites on the Internet
Weak level of visualization of teaching materials, lack of interactivity including hypertext	Interest in multimedia, "visual" communication	Movies and computer games have the second position among the purposes of using computer ranked by students
Step-by-step manuals that presuppose learning by copying the sample, the absence of the original product as a result of the students' work.	Creation of Internet content	Social nets, wiki-sites and forums have the third position among the most frequently visited sites on the Internet. All of them are a platform for a creation of their own content, express their opinion, share things made by themselves. 13% of respondents mentioned the creative activity as a major purpose for using the computer
Students are constrained by one plotline (storyline) in learning, the absence of immersion, problem-solving and decision-making tasks and enough freedom for actions in realization students' learning trajectory.	Multitasks on everything	The sum of hours for different everyday life activities informed by student are about 28 hours a-day.
Weakly realized person-centered approach	Emotionally open	Social nets were ranked as the second position by students
Individual fulfillment and performance of learning results	Teamwork and cooperation	Using computer for communication among students is placed on the third position after learning and entertainment purposes by students

We interviewed students of Faculty of Pre-School and Elementary Education in 2012-2013 academic year. The results confirm last year statistics that quality of teaching materials on KsuOnline was highly appreciated by students, average score was 9.29 in 2011 and 9.16 in 2012 out of 10.

The results of the entrance poll of other faculties in 2011-2013 academic years showed the following trends:

- In 2011, 89% of the respondents owned a computer or a laptop. This academic year, 100% of students are the owners of a computer or a similar device regardless of the discipline.
- 70% (2011), 68% (2012) and 69% (2013) students have an access to the Internet outside of the university – thereby, this rate stays the same.
- However, a number of students who recognize themselves addicted to the Internet has grown from 24% in 2011 to 32% in 2013.
- An interesting result was found by visiting University website by students from different disciplines. 17% of the third year students said they had never visited the university website at the departments where teachers hardly use ICT in teaching. On the other hand the rate was 0 (since 2011 to date) at the departments where most of the teachers regularly use ICT.

The number of students who have a positive attitude towards the use of ICT in education (inter alia, at lectures) has increased (see Fig. 2).

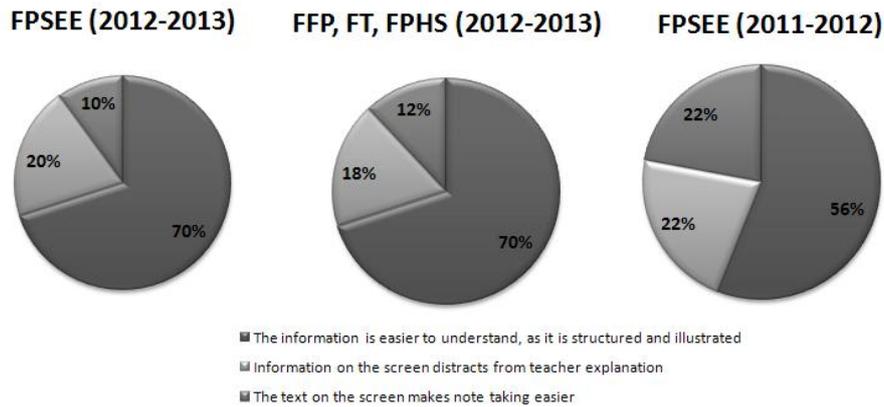


Fig. 2. Students Attitude Towards the Use ICT During Lectures

The purposes of using the computer by students have almost the same rating, regardless of year and the faculty (see Fig. 3).

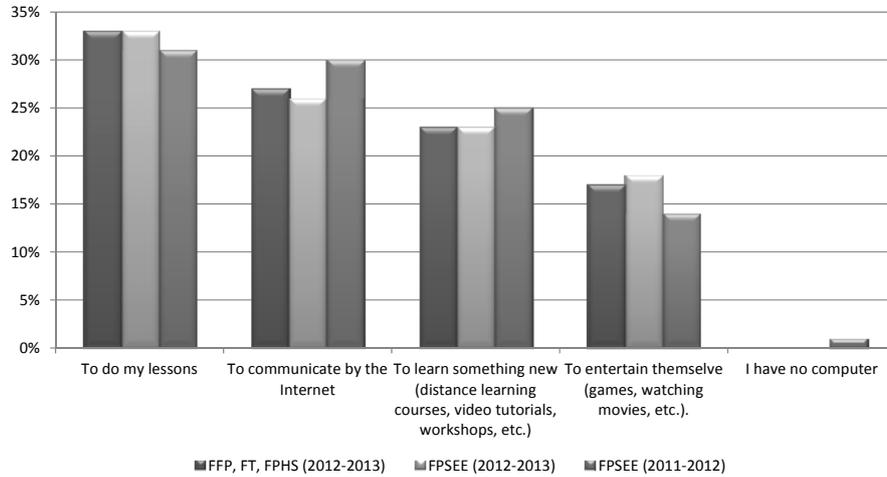


Fig. 3. The Purpose for Using a Computer and Internet by Students

Social net is becoming more popular among services for communication (see Figure), IME and email are also frequently used.

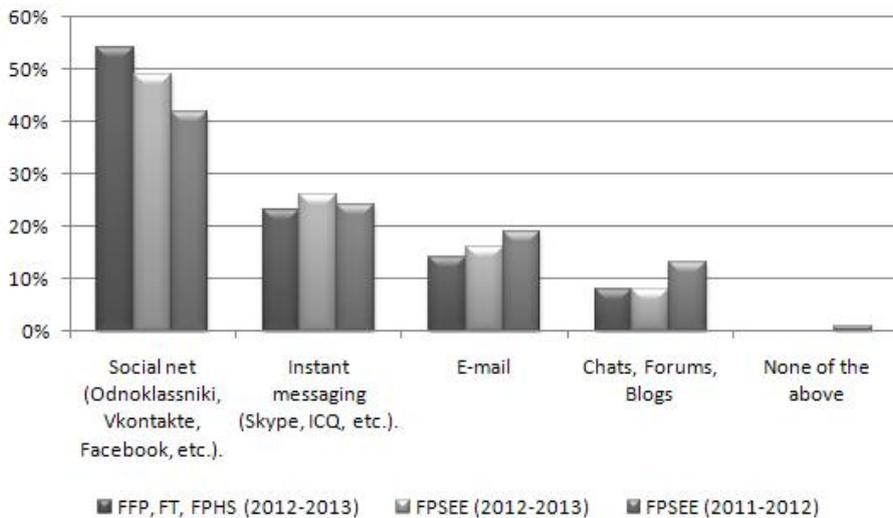


Fig. 4. Student Communication Services

Modern students overestimate their skills for online searching according to Berk. The results of our research have confirmed this fact. The results of entrance poll and test in Informatics are contradictory. The majority of students estimated their level of ICT proficiency as excellent and good. The rate is depicted below.

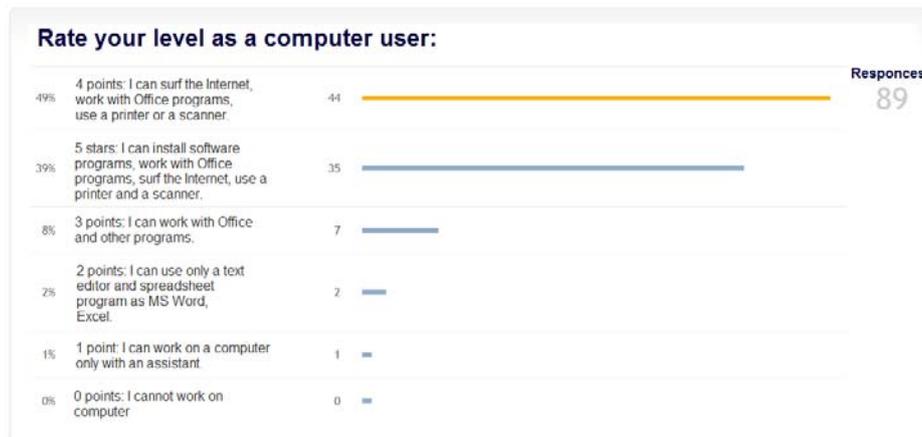


Fig. 5. Results of self-appraisal a computer user by students (FFP, FT FPHS, 2013)

The entrance test contains 45 comprehension questions and is designed by analogy with ECDL test. The aim of testing is to check residual knowledge of school computer discipline. The results showed that only 3.28% of students had a score higher than 4.5, 82% had a score higher than 3 but less than 4,42 points, 62% had from 2 to 3 points and 1.64% scored less than 2 points.

During the recitations, the following problems were identified that are related to organizational issues and students attitudes:

- Students have no skills to manage their time and work efficiently without strict teacher control, regardless deadlines. Only few students uploaded their works in time.
- Not all students use email regularly. Many of them chronically forget their email user names and passwords. Students are confused by various browsers or other version applications. As a result, the login process to email or distance learning system regularly takes up to 7 minutes.
- There is no clear understanding of terms among students. Being able to work with some programs, often they cannot deal with similar ones.
- It is necessary for a teacher to consider different speed of students work while planning lessons. Some students do not regularly use computers. Thus, typing, mouse control, and searching for a command causes a delay.
- Plagiarism. Some students prefer not “to waste” their time to do lessons. They easily copy their colleagues’ results or download similar ones from the net. It is important at the first lesson to highlight the value of students own creative work and punish the students who attempt to copy someone else work. Under strict control, the number of plagiarisms decreases, but does not disappear. In fact, we notice that plagiarism is growing with task level. The reason is that the students desire to get a mark, not knowledge.
- The students do not like reading long instructions. Usually they prefer to ask a teacher or colleagues than to read step-by-step manual. It is also associated with different systems prevailing perception among students. The feature of the course

is that the most of the teaching materials are stored in digital form (in digital form). Creating e-learning course does not require additional financial costs, as opposed to hard copy, faster and easier to edit and update. However, reading from the screen does not give effective results. Perceived only about 40% of the information.

- Lack of collaborative activities: in pairs, groups and teams, social assessment, problem-solving and decision-making tasks. A student presents results of an individual work only to one person—his teacher, and feels subordinated under such conditions. In addition, it is the reason of the absent of the critical view on his own work. This results in frustration and dissatisfaction of the student to any criticism. The teacher comments are perceived in a negative or indifferent way.
- Educators should pay special attention to students feedback about effectiveness of teaching activities and the level of their satisfaction. For example, monitoring and evaluation allow a teacher to obtain an information about teaching incomes and quality, that would improve his work. It should reflect the level the students knowledge. It's vice versa in practice: evaluations become a sign of knowledge / ignorance of the material, the student rating of the group.

The presented requirements to discipline designed in previous work on digital literacy formation [17] have been adapted to Berk's pedagogical strategies. As a result, we have formulated new approaches considering educational needs of the Net students' generation (see Table 2).

Table 2. The approaches of teaching ICT discipline to meet educational needs of generation Net students

The teaching element	Requirements (Description)
Discipline content	<p>A discipline content should reflect current research and encourages students to use new approaches and technologies, highlights current trends in ICT development.</p> <p>Tasks presuppose creative activity, form skills of self-learning and further development. Examples are inspiring.</p> <p>Task execution result is useful, valuable and applicable product in professional practice.</p> <p>All elements of the discipline are focused on future professional activities.</p>
Students motivation	<p>All elements of the course (assignments, surveys, etc.) help students to see themselves in their future profession, particularly in teaching profession.</p> <p>It is important to emphasize interest to the students' opinion, to make possible to their contribution in doing collective projects, for example, to create a bank of teacher's materials;</p> <p>Student wants to get a feedback from his colleagues about his work. Any result of creative task should pass following stages: creation, publication and social assessment with definite criteria.</p> <p>Student wants to evaluate teacher's work too, so a teacher should organize the ways to impact on the discipline development for students, to express their wishes.</p>

The teaching element	Requirements (Description)
Organizational issues	Tasks presuppose collaboration, facilitate communication and interaction to develop personal aspect of student and help to realize his individuality. Clear structure, planning, to do list and deadline system. Active use of formative assessment techniques. Ice-breaking, team-building and communicative exercises at the beginning and at the end of the lesson to form communicative skills, values, relationship. Use no more than two new environments at the lesson. Teacher should consider in the selection of services to work at the lesson: registration absence or its simplicity, functionality easiness, necessity to install additional software opportunities to use in profession.

The main aim of updating the discipline was to help future teachers to create and organize their own learning space in the Internet. Therefore online interactive services that can be used for communication and teaching pupils were included (creating word clouds, mind map, open online documents, site, etc.). We also considered Berk’s strategies while designing the discipline. The updated version of "Information Technology" discipline was taught at the Faculty of Foreign Philology (FFP) - 104 students, Faculty of Translation (FT) – 61 students, Faculty of Psychology, History and Sociology (FPHS) – 28 students.

Table 3. The implementation of Berk’s teaching strategies in the course "Information Technology"

Characteristic of Net Generation	The implementation of educational strategy
Tech savvy	The virtual discipline environment was created with the system of distance learning ksuonline.ksu.ks.ua located on a MOODLE platform. This system allows developing a course with such elements as glossaries, wikis, multimedia clips, presentations, tests, blogs, forums, etc.
Teamwork and cooperation	Some tasks are complex and presuppose collaboration, while their execution the cognitive interpersonal communication and interaction of all participants progress. An important stage is to prepare a group of students to work together. To ensure about students’ readiness for cooperation teacher should arrange Ice braking, team-building and communicative exercise (up to 5 minutes) at the beginning and at the end of lesson. Teacher, who works in the computer classroom, usually recognizes his students primarily "from the back." Therefore, such exercise facilitates personal contact with the group, which is especially important while teaching short disciplines.
Interested in multime-	The discipline content was developed and designed considering

Characteristic of Net Generation	The implementation of educational strategy
dia, "visual" communication	students' interest in "visual communication" and multimedia. Video-clips and presentations were added to each theme. The tasks and manuals contain minimum of text information and maximum of graphics and illustrations. Teachers included such elements as blogs and forums.
Rely on search engines. Multitasks on everything	In adding to use search engines during the course students active work with Google services as Google.Drive, Google.sites, etc. This satisfies the interest in creating online content and allows simultaneously work with several documents. It's also an efficient tool of collective activity organization.
Retention in the profession	Organization of students' activity is a process of consistent modeling of professional activity of specialists under learning conditions. The students of teaching specialties create educational games, tests, documents, tables that are useful in their professional practice. The results of the work are evaluated not only technical element, but also educational one. Also such task as creating a presentation "My choice" (students aimed to describe situation of their professional choice, analyze their present achievements and capacities, goals and dreams, ways to attain them) and collective writing of mind map "Modern teacher" were added.
Emotionally open	Tasks are person-centered and presuppose creation unique results that will describe student's own lifestyle, attitude and etc. This approach helps decrease the quantity of plagiarisms and contributes to the development of their creative abilities. Each student's work passes through formative assessment: self-appraisal, social assessment, teacher's assessment.

Such tasks as making clouds of words, mind maps, playing learning games are relevant to professional and educational discipline orientation. In addition, they presuppose creative activity and can be easily adapted to a group work.

Poll Results of 2012-2013 academic year at Faculty of Pre-School and Elementary Education are comparable with poll results published one year earlier. The quality of teaching materials on KsuOnline highly appreciated by students, average score was 9.29 in 2011 and 9.16 in 2012-2013 out of 10. The novelty of lectures was assessed as 7.59 in 2011 and 6.84 in 2012-2013, the novelty of practical tasks had 7.64 and 7.39 respectively.

The students' preferences about the most interesting and useful for future professional tasks have changed. In both of these categories the task "Creating didactic game" leads. Making a poster "It's interesting to know," creation of "Site Class" and creation of online poll follow in the rating. These tasks we preserved in the updated version of the discipline "Information Technology". As a result, new version of "Information Technology" discipline has the following structure:

Lesson 1. Registration on KSUOnline, taking an entrance poll and test. Presentation "My choice" with the following structure:

1. My strengths and weaknesses (academic, creative, personal, individual aspects).

2. My goals, objectives and deadlines (to 2020 year).
3. My completed tasks.
4. A letter from You yourself in 2020.

Lesson 2. Creating a class site on Google Sites (Students add, delete pages, edit them and insert pictures and text, change site's design). Taking a poll "NET Gener profile Scale".

Lesson 3. Creating a poll on Google Drive and inserting it to the site.

Lesson 4. Creating a cloud of words (Tagxedo service), publishing pictures with a cloud of words and writing an assignment for the pupils on the "Homework" site page. Changing assignments with the word clouds and its execution.

Lesson 5. Creation educational games with triggers or hyperlinks in MS PowerPoint.

Lesson 6. Social expertise of the games (organized with Google Drive – a spreadsheet). Watching the movie "The image of the modern student" by Michael Wesch (Kansas State University, 2007). Creating a collective essay "How I like to learn" in online document on Google drive.

Lesson 7. Co-creation of mind map "Modern teacher." Taking final poll and test.

Making a poster "It's interesting to know" and contributing to wiki page "Communication in the network" are for independent work.

Returning to the problem of assessment, it should be noted that in presentation "My Choice" 68 students described their school successes and achievements considering them as the next step in their professional development, but some students evaluate these achievements as formal, not real:

"To the moment of leaving school I had about 60 letters of honor in my "collection". But I understand they only have formed my ability to learn and now mean nothing".

It's necessary to change assessment system shifting the emphasis from control to formative function for strengthen the motivation. In addition it's necessary to vary teacher's assessment with self-assessment and social assessment due to teacher's criteria. Students can determine the quality of the work due to criteria and assess technical realization, structure, relevance to age of pupils, subject, design, quality of illustrations, literacy, etc. This forced students to think about the quality of their work while creation it.

Social expertise has stimulating, diagnostic and formational functions. A wish to get social appreciation motivates to create high-quality bright individual works. Such way of assessment forms critical perception of the information. A student with low self-esteem at first is afraid that his group mates will assess not his work but his person, but it has never happen, all students are aimed to be objective and independent judges.

The social aspect is a priority for a new generation. They tend to have a relationship regardless of the field of interaction. Education is not an exception. If the teacher does not build the relationship consciously, it usually takes the worst form. A teacher should know that a grade is no longer a sufficient motivation for the positive attitude of the student to the discipline, especially in the pass-fail system. It is clear that students often associate the discipline with teacher's personality. The high quality

executed tasks show not only an attempt to get a high grade but sympathy to a teacher. In computer disciplines social aspect is often nearly absent or poorly developed: most of the time students work individually on the computers.

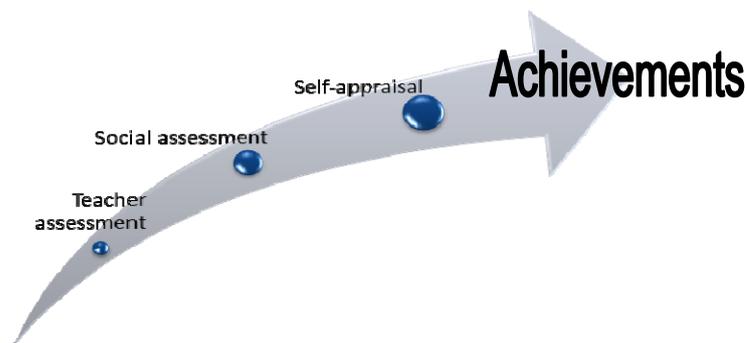


Fig. 6. Increasing students' achievements by the method of formative assessment

The times of detached teacher have passed. Today's successful teacher can easily contact with the audience in a short time, acts as a partner, friend and a leader.

During the course, we found that the most third-year students have lack of communication. For example, at first lesson after the game "Introduction", we asked all students to tell us about their three small victories. Such "victories" were often met with utter surprise of the group. It was uncovered that the students' ability to learn is directly connected with their sense of self-confidence and positive mood.

Obviously, a success of training approach strongly depends on a teacher personality and his skills in training and teambuilding. Therefore, the authors of this paper have selected games and exercises that do not require special teacher's skills and knowledge or any additional devices.

Training elements in students' groups of different teachers' specialties and years of study had different reaction. Some students expressed stunning and rejection: "Why should I do that? It is not serious." despite the fact that they had finished Innovative Teaching Methods and Technologies discipline, training exercise was an extraordinary situation. One of the teachers said describing her experience: "Third-year students look so serious, that sometimes I feel scared to come to the classroom, as if teachers and students have switched their roles...". We find it particularly important to use the elements of training and communicative techniques working with students of teaching specialties. Present students are future teachers who are taught traditionally. In several years, they'll copy one of teaching styles they have seen. It is important to implement innovative techniques and give to students a chance to test them in practice.

Another important improvement in the organization of the discipline was using of open online document on GoogleDrive among teachers. Each of the teachers contributed to planning, selection of training exercise and other notes. After each lesson, teachers made a record in "a discipline online diary" to describe in free form results of the lesson, such as the most common student mistakes, teaching and

technical problems, positive situations, questions from the students, etc. Keeping a discipline diary had a great success. Teaching has become more effective and convenient, the level of awareness and collaboration among teachers has risen, so now teachers create and use it in other disciplines.

4 Results and Discussion

Developed approaches allowed us to improve a range of disciplines including "Information Technology", which is taught for students of all teacher specialties. We also applied them to following disciplines: "Introduction to Information Technology" (for future teachers of elementary school and Computer Science, the 1st year of study), "Fundamentals of Computer Science and Applied Linguistics" (translators, the 2nd year of study), and "Office Computer Technology" (programmers, the 1st year of studies).

Students expressed their positive attitude verbally in the classroom and several students sent e-mails with gratitude after finishing the discipline.

The results of the final poll confirmed their appreciation.

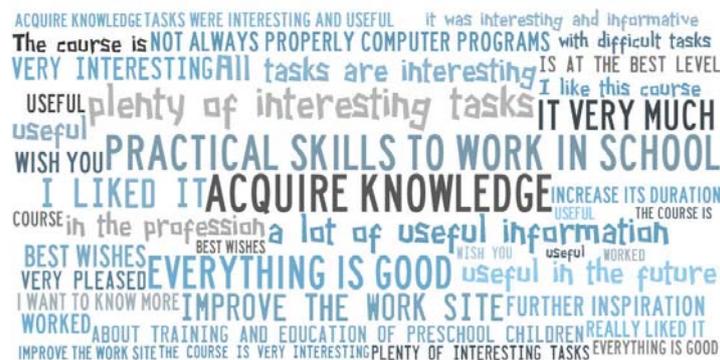


Fig. 7. Word cloud composed of students' comments about IT discipline

5 Conclusions and Outlook

Thus we have obtained the following results:

1. Discussed and analyzed characteristics of generation Net students;
2. Highlighted the contradiction between the characteristic of today's students and traditional ways of ICT teaching;
3. The approaches to resolve these contradictions are found and implemented in teaching practice.

In the future we plan to expand the list of pedagogical specialties to which the updated version of the course "Information Technology" will be taught and improve other disciplines according to the proposed approaches.

References

1. ICT in Higher Education in CIS and Baltic States: State-of-the-Art, Challenges and Prospects for Development. Analytical Survey, 6, GUAP, St Petersburg (2009)
2. Berk, R. A. Teaching Strategies for the Net Generation. *Transformative Dialogues: Teaching & Learning Journal*, 3(2), 1–23 (2009)
3. Cashmore, P.: Stats Confirm it: Teens don't Tweet. Nielsen NetView Audience Measurement Survey, July 2009, <http://mashable.com/2009/08/05/teens-dont-tweet> (2009)
4. DeAngelo, L., Hurtado, S. H., Pryor, J. H., Kelly, K. R., Santos, J. L., Korn, W. S.: *The American College Teacher: National Norms for the 2007–2008 HERI Faculty Survey*. Higher Education Research Institute, UCLA, Los Angeles (2009)
5. Frand, J. L.: The Information Age Mindset: Changes in Students and Implications for Higher Education. *EDUCAUSE Review*, 35, 15–24 (2000)
6. Greenberg, E. H., Weber, K.: *Generation We: How Millennial Youth are Taking over America and Changing our World Forever*. Pachatusan, Emeryville, CA (2008)
7. Horrigan, J. B.: *Home Broadband Adoption*. Washington, DC: Pew Internet and American Life Project (2006)
8. Horrigan, J. B., Rainie, L.: *Internet: the Mainstreaming of Online Life*. Washington, DC: Pew Internet and American Life Project (2005)
9. Junco, R., Mastrodicasa, J.: *Connecting to the .Net Generation: What Higher Education Professionals Need to Know about Today's Students*. Washington, DC: Student Affairs Administrators in Higher Education (NASPA) (2007)
10. Kridl, B.: *The condition of education*. National Center for Education Statistics (NCES), Washington, DC, U.S. Department of Education, Office of Educational Research and Improvement, National Center for Education Statistics (2002)
11. Ostrow, A. Stats: Facebook Traffic up 117%, Veoh Soars 346%. Nielsen Net Ratings, August 2007, <http://mashable.com/2007/09/13/nielsen-august> (2009)
12. Pryor, J. H., Hurtado, S., DeAngelo, L., Sharkness, J., Romero, L. C., Korn, W. S., Tran, S.: *The American Freshman: National Norms for fall 2008*. Higher Education Research Institute, UCLA, Los Angeles (2009)
13. Sax, L. J., Ceja, M., Terenishi, R. T.: Technological Preparedness among Entering Freshman: the Role of Race, Class, and Gender. *Journal of Educational Computing Research*, 24(4), 363–383 (2001)
14. Tapscott, D.: *Growing up Digital: How the Net Generation is Changing your World*. McGraw-Hill, NY, (2009)
15. Behrstok, E., Clifford, M.: *Leading Gen Y Teachers: Emerging Strategies for School Leaders*. TQ Research&Policy BRIEF, Washington, DC, USA (2009)
16. Petuhova, L.E.: *Theoretical and Methodological Background of Informational competencies Formation of Elementary School Teachers*. Doctoral Dissertation, Specialty 13.00.04 - Theory and Methods of Professional Education. The South Ukrainian National Pedagogical University named after K.D. Ushynsky, Odesa (2009)
17. Kushnir, N., Manzhula, A.: Formation of Digital Competence of Future Teachers of Elementary School. In: Ermolayev, V. et al. (eds.) *ICT in Education, Research, and Industrial Applications*. Revised Extended Papers of ICTERI 2012, CCIS 347, pp. 230-243, Springer Verlag, Berlin Heidelberg (2013)

Pedagogical Diagnostics with Use of Computer Technologies

Lyudmyla Bilousova¹, Oleksandr Kolgatin¹ and Larisa Kolgatina¹

¹Kharkiv National Pedagogical University named after G.S.Skovoroda, Kharkiv, Ukraine

lib215@list.ru, kolgatin@ukr.net, larakl@ukr.net

Abstract. The technology of the automated pedagogical diagnostics is analysed. The testing strategy, oriented for pedagogical diagnostics purpose, and grading algorithm, which corresponds to Ukrainian school grading standards, are suggested. "Expert 3.05" software for automated pedagogical testing is designed. The methods of administration of the database of the test items are proposed. Some tests on the mathematical topics are prepared with "Expert 3.05". The approbation of these tests in the educational process of Kharkov National Pedagogical University named after G.S.Skovoroda is analysed.

Keywords. E-learning, Diagnostics, Test

Key terms. InformationCommunicationTechnology, Teaching Process

1 Introduction

Pedagogical diagnostics is the integral part of adaptive E-learning courses. The unconditional quality of testing is its high informative abilities. However, in practice the large part of the test information is often not used. Computer technologies give us possibility to organize the qualitative pedagogical diagnostics at new level. Modern automated systems which can be qualified as expert systems are capable to supply comprehensive algorithms of testing and analysis of the test results. Testing with use of computers allows a teacher to obtain the summary characteristics of knowledge and skills of the pupils' group and to use this information to choose the teaching methods. A study of such algorithms is a wide field of the scientific work. Therefore, the aim of our paper is to design methods of the pedagogical diagnostics, which satisfy following demands:

- Different forms of the intellectual activities of an examinee are attracted in process of testing;
- The automated system of the pedagogical diagnostics ensures its diagnostic abilities at wide differences of the examinees mastering;
- Processing of the test results provides maximum information for an examinee and a teacher to correct the educational process.

2 Objectives

The first stage of pedagogical diagnostics organising is a construction of an idealised pedagogical model that is allocation of basic elements of knowledge and skills, as well as detecting the level of its mastering.

The second stage represents creation of the problems system which covers all elements of knowledge and skills and all levels of their mastering.

We cannot design test as a system of test items of equal difficulty, in spite of recommendation of the classic test theory. Such approach gave the best tests for discrimination of examinees into several groups. However, the test with equal items has low validity for examinees with bad mastering because of guessing answers. Validity of such a test is also low for good mastering examinees because of lack of attention. Therefore, it is certainly necessary to include problems of different difficulty to the test.

How to design a test item of advanced difficulty? What is difficulty? Why the most of examinees do not solve some problems?

We cannot use problems of the reproductive level as items of advanced difficulty. There are not difficult facts and easy facts. Our educational process should be organised to provide steady knowledge of all compulsory facts. If the most of examinees do not know some compulsory facts, it means that we should correct our teaching. We are against using items which correspond to facts that are fragmentary studied and which are not basic for the tested topic. Therefore, all problems corresponding to the reproductive knowledge must have equal difficulty.

Someone can increase the difficulty of an item by combining several operations in this item. Such approach leads to increasing influence of lack of attention on the test results, as well as to necessity of using weight coefficients and to decreasing of the measuring accuracy of the test. We are also against using problems which correspond to facts that are fragmentary studied and do not form basis of the topic being tested.

In our opinion, the item of advanced difficulty should be connected with use of more difficult, not reproductive kinds of the intellectual activities [1], [2].

Full and high-qualitative pedagogical diagnostics should be built on the system of test items of all levels: reproductive and productive. By analogy with levels of educational achievements [2], which are standardised by the Ukrainian Ministry of Education and Science [3] we propose the following levels of the test items:

1. Initial level - it is the very simple problems which assume the reproductive character of the student's activities, mainly distinguishing. The difficulty index of these test items is about 1, the most of the examined students execute these items correctly.
2. Average level - it is the problems which assume the reproductive activities, these problems cover all basic facts and unary skills according to curriculum. A database of items of this level is designed the most naturally. According to the Ukrainian standards [3], the student can continue education, if he (she) knows not less than 50% of compulsory facts determined by curriculum. Therefore, by linear estimation, average index of difficulty must be near 75% for the reproductive items.

3. Sufficient level - these items assume the examinee applies his knowledge and skills for solving problems in standard situation.
4. High level - these test items are practical problems which assume executing of new algorithm, carrying knowledge into new, non-standard situation, etc. These items can lose creative nature, if the method of its solving was explained in the process of learning. Therefore, the database of the items of the level 4 requires continuous analysis and modernisation.

We propose the vector processing of the test results – separate calculation of the score for items of every level. It allows to avoid the use of the artificial weight coefficients and to provide the comprehensive algorithm of adaptive strategy of testing and grading. We also propose the separate processing of the results for the test items according the elements of knowledge and skills.

Using computer for test administration allows to analyse the examinee's results directly in process of testing and to suggest an examinee the items, which mostly correspond to his (her) level of educational achievements. Such approach is often called adaptive or quasi-adaptive testing [4].

3 Model and Algorithm

A choice of the items level for start of testing is an important question of the adaptive strategy. Testing usually starts from the simplest items. Such approach provides decreasing of psychological discomfort and creates the atmosphere of competition, the feeling of growth according to complication of the problems. Taking into account this consideration we propose to start testing just from items of the level 2 which are the simplest for the examinees that will obtain positive grade.

There is additional argument to choose the level 2 as the start level of testing. The test items of the level 2 reflect the compulsory facts of the study topic; these problems cannot be excluded from the test process. It is not worthwhile to start test from the items of the level 3, because productive and, especially, creative problems are based on sufficiently wide spectrum of knowledge, and it is not always possible to detect, which exactly element of curriculum is not mastered by an examinee. The items of the level 1 are intended for students whose mastering is not satisfactory; therefore, there is no need to suggest these items to all examinees. Our testing strategy and algorithm of grading are presented on the fig. 1.

Here are some comments to fig. 1. The testing starts with the items of the level 2. An examinee solves the compulsory minimum of the items on the level 2, automated system calculates S_2 - his (her) score on the level 2 and estimates the error of the score. If accuracy is enough, the automated system chooses a grade or increases the level of items, which are being suggested to the examinee. Otherwise, the items of the level 2 are being suggested to the examinee until the accuracy become satisfactory. It should be underlined that accuracy depends not only on the number of items, but it depends on the individual test score [5]. The necessary accuracy is also conditioned

by the differences between the test score and the key points for decision about grading or a rise of a level.

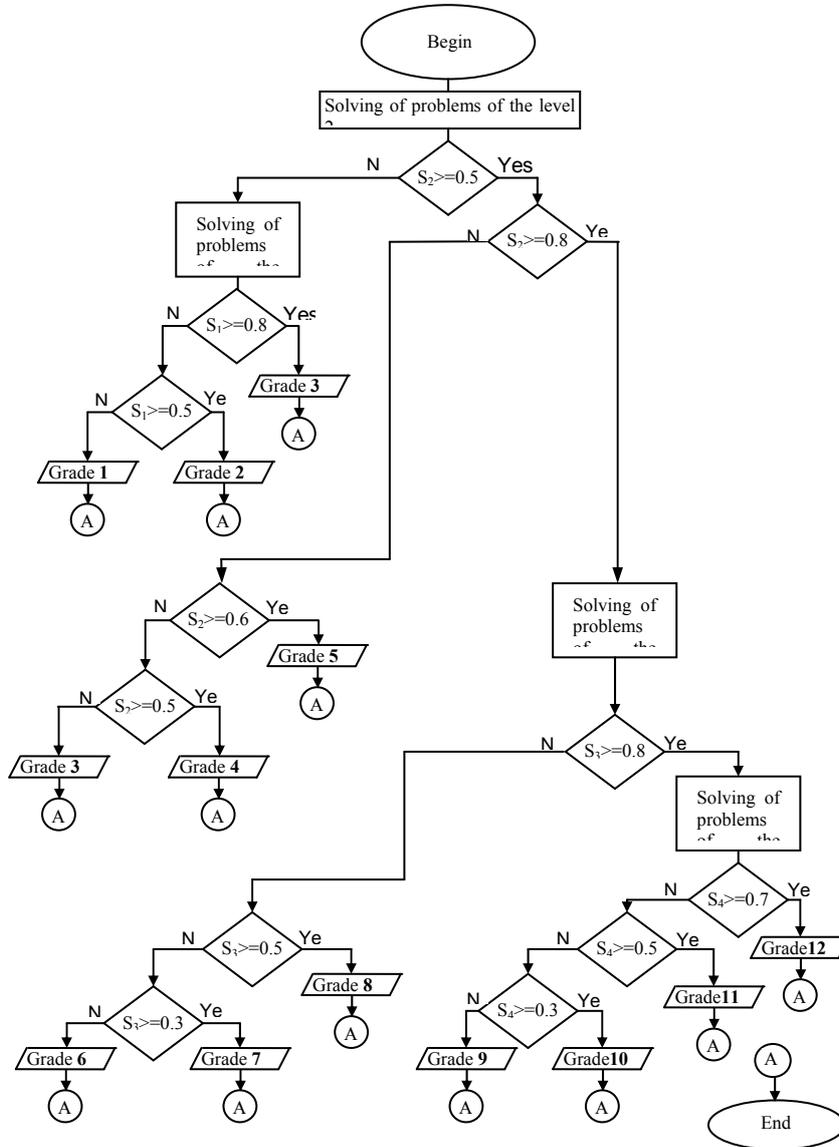


Fig. 2. Grading algorithm

Testing on the levels 3, 4 and 1 is carried out by analogy with level 2 with scores S_3 , S_4 and S_1 accordingly.

Such algorithm of testing improves accordance between the level of items being suggested and the level of the examinee's mastering. Influence of a lack of attention on the test score for examinees with an excellent mastering is decreased. The exami-

nees with bad mastering solve easy problems, which correspond to the most important parts of the educational content. The psychological discomfort, connected with constantly incorrect answers, is excluded, but such easy items give a possibility to determine the structure of the knowledge and skills, as well as to distinguish examinees, who have not mastered the compulsory minimum according to curriculum.

In any case the result of the pedagogical diagnostics will be more careful in comparison to testing without special selection of items.

4 Software

The computer support of offered technology is provided with information system "Expert 3.05" designed by us as a distributed database in the MicrosoftAccess2003 environment. The important advantage of our information system in our opinion is a modular principle of its construction which allows the author of the test items (probably, with the help of the programmer) to create and to add to the database new forms of the test items. The central database of information system is the test items database. The test items are grouped by topics for convenience of viewing. The elements of an educational content are picked out in each topic. To each element the author provides the comment for a student who has not mastered this content. Some blocks of the test items of a different level of difficulty are offered to verify student's mastering in each element of an educational content. The author specifies for the every block such parameters: a test items level (0-4), a weight factor and maximum time of exposition of one item. All items of the block should be of one type, that is, the identical dialogue form. The student will be offered one or several items from each block by a casual choice in a process of testing. Quantity of items blocks and filling these blocks are determined by required quality of diagnostics.

The database, which contains the information on the answers of each examinee on each item, is formed by results of testing. This database includes such fields: the code of the test item, the correctness of the given answer, the level of the item, probability of casual guessing of a correct answer, time of item solving. The additional service information (time and date of testing, examinee's grade etc.) is also being stored.

The examinee receives (fig. 2) the diagnostic data on each element of knowledge; chart, which reconstructs a structure of his (her) knowledge; recommendations for independent work. The author receives the statistical analysis of each item difficulty and discrimination, correlation with the test score (grade). The diagram shows dependence of item difficulty on the examinee's test score (grade) and is very useful (fig. 3). The author has an opportunity to generate database query by means of Access environment and to pass the data for the further analysis in spreadsheets.

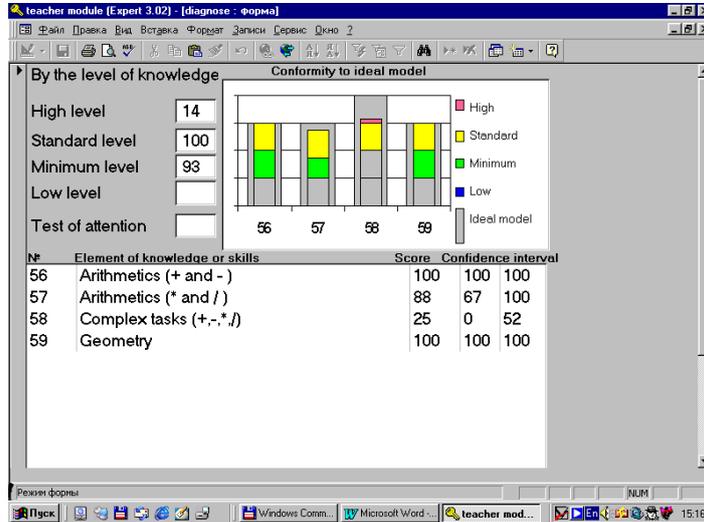


Fig. 3. Diagnostic data for the examinee

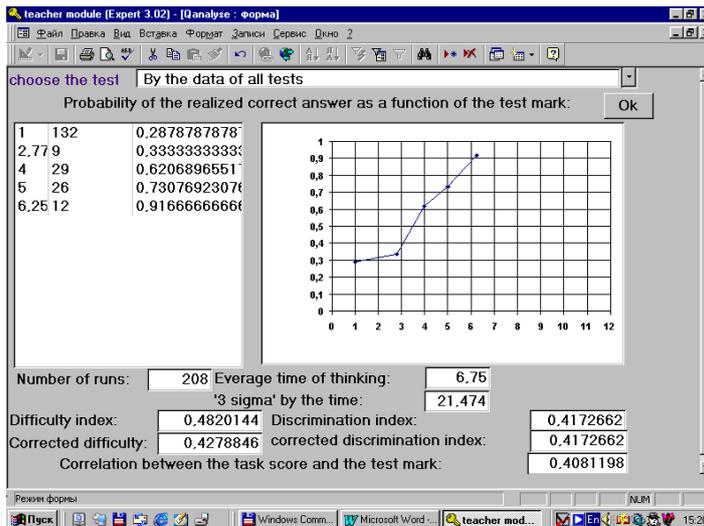


Fig. 4. Statistical analysis of a test item for the author

5 Experience of Diagnostics with “Expert 3.05”

We have prepared the system of the test items with "Expert 3.05" on some courses: “Mathematical methods in psychology”; “Theoretical basis of informatics”; “Architecture of a personal computer”.

Now we are able to start the third stage of the preparation of the system of the pedagogical diagnostics - approbation and verification of the test items.

Our technology of verification is based on the requirements of the Standard of the Ukrainian Ministry of the education and science [6] and takes into account features of the automated pedagogical diagnostics.

The analysis of the test begins with detecting of a level of educational achievements of the students based on the expert's rating, for example, it may be a traditional examination. The complete verification procedure assumes that the experts determine such rating irrespective of the verifying test. Approbation should be organised with enough number of examinees to guarantee sufficient number of answers for any test item. Determination of the student's rating by experts cannot be organised so often, as it is necessary for constant updating of the test items database. Therefore, for the current verification it is possible to offer detecting of a level of student's educational achievements with the help of the same automated system of pedagogical testing and to check a correlation of the separate item with the test score. In such a case, the approbation data are accumulated continuously, including the independent work of the students with the automated system. Validity of the automated system of testing as a whole is checked through comparison of integrated results of testing with results of other kinds of the control: interview, examination, execution of practical works etc. For maintenance of reliability of the current verification, the automated system does not include into the analysis the answers of not registered examinees: teachers and other users, whose names are not in the lists of the students groups. The answers of the test pass, which has not been finished, are not analysed too. The answers are not taken into account, if the time of its execution is smaller, than it is necessary for acquaintance with the text of the problem. There is an opportunity to specify additional conditions of selection of the valid answers with use of the Access environmental (for example, date and time of testing, educational group, variant of the test etc.).

After distribution of the students according to their educational achievements, the conformity of a level of the test item and its empirical index of difficulty is checked. According to the requirements of the Ukrainian educational standards, the students with an average level of educational achievement (the grade of 4 on the grading scale with 12 grades) "... knows about half of educational content, is capable to reproduce it, to repeat after the model the certain operation..." [3]. Just the items of the level 2 in our classification have such contents. Thus, the difficulty index of the items of the level 2 can not be less than 0.5 for such students and we consider it to be within the range 0.5-0.9. It is necessary to note that such a range of difficulty is not convenient from the viewpoint of improvement of test statistical parameters. However, the items of the level 2 represent a set of educational content facts, which are obligatory for mastering. Therefore, the problems author cannot change its difficulty without changing the curriculum.

Thus, for the items of the level 2 on the sample of students with an average level of educational achievements we have the following algorithm of analysis of the item quality:

- An item does not require correction, if its difficulty index within 0.5-0.9, the discrimination index is higher than 0.25, that is, discrimination index satisfies the requirements of the standard [6] (fig. 4.).

- The item difficulty index is more than 0.9. This item should be analysed with the help of the diagram of dependence of difficulty from educational achievements of the students. It should be determined, whether this item has the discrimination ability for the students with the initial level of educational achievements, accordingly, this item level should be changed (fig. 5). Otherwise, this item should be removed from the test.
- The difficulty index is less than 0.5, it is necessary to analyse the content of the item, such situations are possible:
 - The item is not reproducible, its discrimination ability satisfies the requirements of the standard. It is necessary to increase the item level (fig. 6).
 - The item has low discrimination ability for all students; it signifies that some mistakes in the formulation of the item take place. This item should be removed or corrected (fig. 7). There is a possibility of the situation, when all experts agree that the problem is correctly designed and satisfies the curriculum, in such a case mastering of the students should be checked by some another method and, may be, the quality of educational process should be analysed.

The best range of difficulty index for items of levels 1, 3, 4 will be within 0.5-0.6 (allowable 0.3-0.7) on the sample of the examinees of appropriate level of mastering. The analysis of these items on a difficulty is carried out by analogy with level 2.

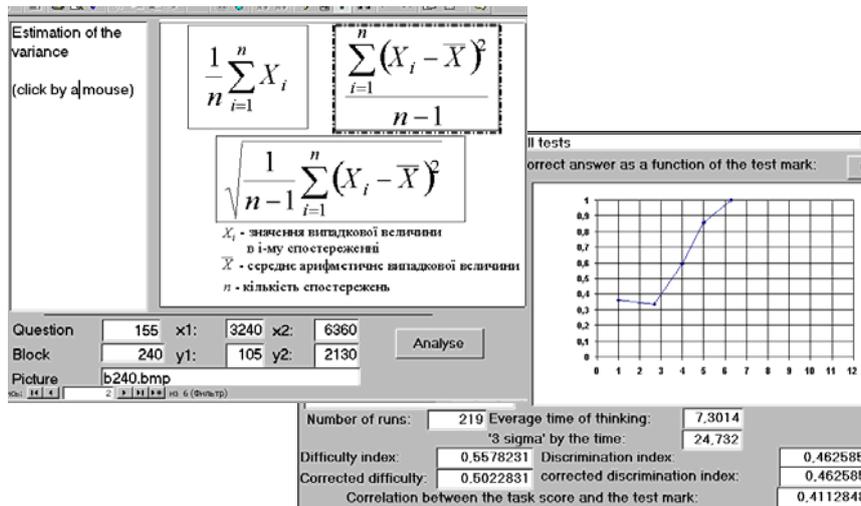


Fig. 5. A typical item of the level 2

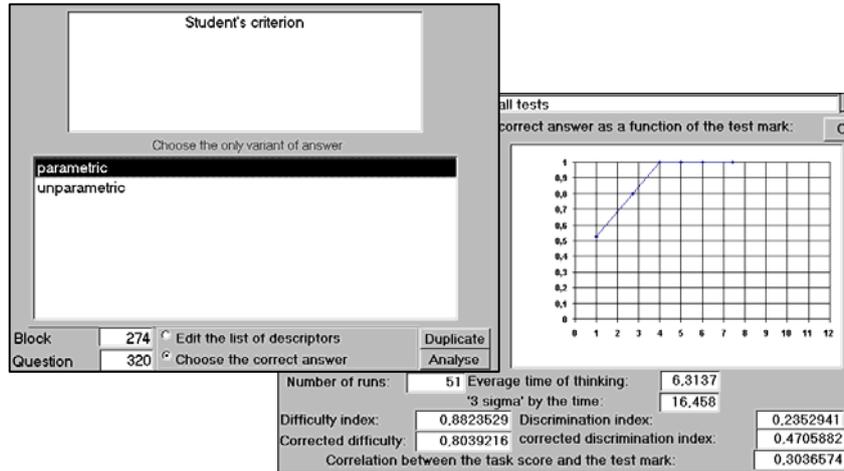


Fig. 6. A typical item of the level 1

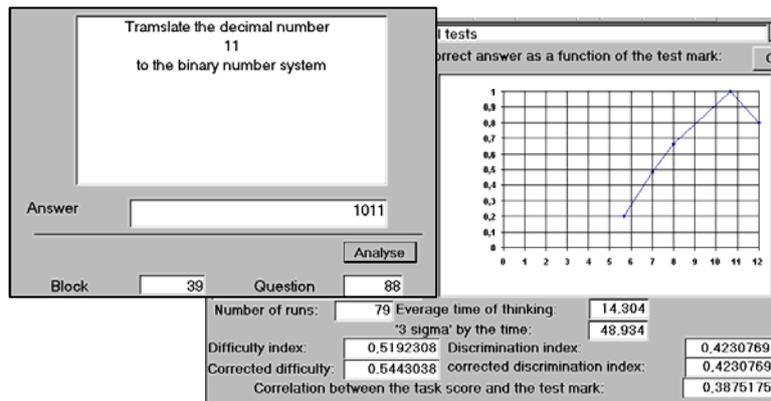


Fig. 7. A typical item of the level 3. It is not a reproductive problem, if the students do not learn the table of the binary codes for numbers until 15.

After finishing three stages of preparation the system of pedagogical diagnostics is ready for practical use. The stage of practical application of the system combines procedures of testing and statistical processing of the obtained results including the interpretation of the results for students, teachers and authors of the test problems. The expert system of pedagogical diagnostics needs continuous modernisation of its database. Naturally, it requires returning to previous stages of the work with the system.

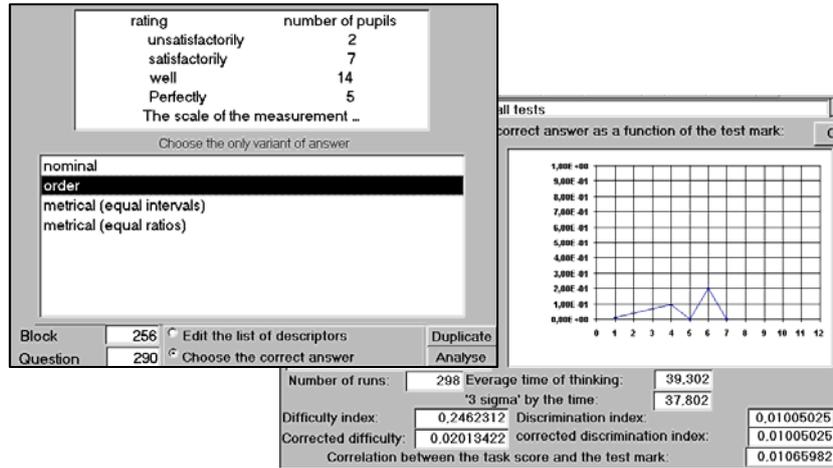


Fig. 8. An unsuccessful item

The “Expert 1.01-3.05” software is used in Kharkiv National Pedagogical University named after G.S.Skovoroda since 2001. Here are some latest results on approbation. The test on mathematical methods of statistical analysis of the pedagogical diagnostics data was suggested to future teachers of informatics, mathematics, chemistry as an element of courseware "Information systems in pedagogical activities" in 2012-2013 academic year. The purpose of testing was the self-diagnostics of students. So, the students could pass the test many times, studying the problem elements of the learning content and improving their results. We took into account the best results of testing and compared its with the examination results. The Pearson correlation was 0.7 at sample of 51 students and we can consider it as the test validity.

The test results gave us possibility to study the structure of students’ knowledge at basic questions of statistical analysis of the pedagogical diagnostics data (fig. 8). The error estimation for the data on fig. 8 was evaluated as a half of the 95% confidence interval

$$\Delta y = \frac{1.95s}{\sqrt{n}},$$

where s is the estimation of the standard deviation and n is the number of test items of the given learning element, which were passed by students. The errors are different for every point on fig. 8, because the number of test items on various elements is different, so we show the ranges of errors in table 1.

The results (fig. 8) show that problems of choosing the scales for the pedagogical evaluations are the most difficult for students. The problems of reproductive level, where student should to choose the method or formula for estimation of some parameters of statistical distribution, are the easiest. But on the productive level, when students should explain the influence of the values and number of variants in a sample on the estimated parameters, such problems are the most difficult.

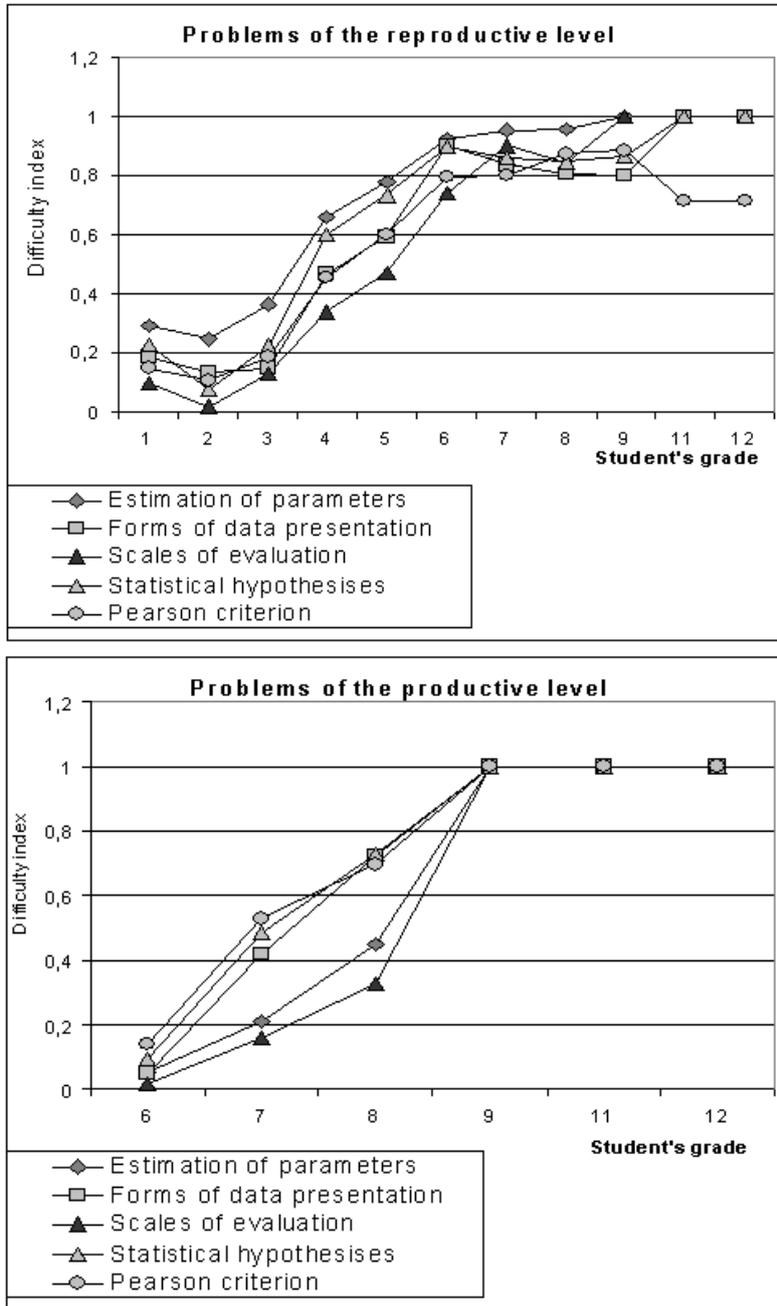


Fig. 8. Difficulty index (probability of correct answer) as a function of the student's grade for problems of various elements of the learning content

Table 5. Errors of difficulty index estimation at different student's grades

Student's grade	Estimation error
1	0.01-0.04
2	0.04-0.1
3	0.02-0.05
4-5	0.01-0.07
6-7	0.02-0.16
8	0.03-0.18
9	0-0.4
10-12	0-0.6

6 Conclusions

1. New comprehensive algorithm of testing and grading is suggested. This algorithm takes into account possibilities of the computer technologies and requirements of Ukrainian standards.
2. The automated system of the pedagogical diagnostics is designed.
3. The methods of the items database administrating is proposed and used in practice in the educational process of the Kharkov National Pedagogical University.

References

1. Bloom, B.S.: Taxonomy of Educational Objectives. Book 1: Cognitive Domain. Longman, Inc., New York (1956)
2. Bespalko, V.P.: Basis of the Pedagogical Systems Theory: Problems and Methods of Psychological and Pedagogical Providing of Technical Teaching Systems, Voronezh University Press, Voronezh (1977)
3. Criteria of Grading of the Educational Achievements of Pupils in the System of Secondary Education. Education of Ukraine, 6 (2001)
4. Zaitseva, L.V.; Prokofyeva, N.O.: Models and Methods of Adaptive Knowledge Diagnostics. Educational Technology & Society, 7, http://ifets.ieee.org/russian/depository/v7_i4/pdf/1.pdf (2003)
5. Kolgatin, O. G.: The Statistical Analysis of the Test with Different Forms of Items. Means of Teaching and Research Work, 20, KhSPU, Kharkiv (2003)
6. Means of Diagnostics of a Level of Educational and Professional Training. The Tests of the Objective Assessment of a Level of Educational and Professional Training. Order of the Ministry of Education and Science of Ukraine, № 285, 31 July 1998 (1998)

The Use of Distributed Version Control Systems in Advanced Programming Courses

Michael Cochez, Ville Isomöttönen, Ville Tirronen and Jonne Itkonen

Department of Mathematical Information Technology
University of Jyväskylä
P.O. Box 35 (Agora), 40014, Jyväskylä, Finland

{michael.cochez, ville.isomottonen, ville.tirronen, jonne.itkonen}@jyu.fi

Abstract. Version Control Systems are essential tools in software development. Educational institutions offering education to future computer scientists should embed the use of such systems in their curricula in order to prepare the student for real life situations. The use of a version control system also has several potential benefits for the teacher. The teacher might, for instance, use the tool to monitor students' progress and to give feedback efficiently. This study analyzes how students used the distributed version control system Git in advanced programming related courses. We also have data from a second year course, which enables us to compare between introductory level and master's level students. We found out that students do not use the system in an optimal way; they do not commit changes often enough and regard the version control system as file storage. They also often write commitmessages which are meaningless. Further, it seems that in group work settings there is usually one dominant user of the system.

Keywords. Programming Education, Version Control System, Git

Key terms. ICTTool, TeachingProcess, ICTEnvironment, Technology

Introduction

Version control systems (VCSs) have a decades-long history in professional software engineering with early systems like Source Code Control System (SCCS) and Revision Control System (RCS) developed in the seventies and eighties respectively. These pioneering systems only supported storage of the versions on the file system, while later systems also allowed for remote and mostly centralized storage of the versions. The most well-known centralized systems are Concurrent Versions System (CVS) and Subversion (SVN). Currently, there is a trend towards the use of distributed version control systems (DVCS) where each user has a local copy of the repository which can be synchronized with other repositories. Systems such as Git and Mercurial exemplify this type of present-day decentralized technology. These DVCSs enable flexible change tracking, reversibility,

and manageable collaborative work, which are valuable for both small and large projects.

There are many arguments for incorporating VCSs into an educational setting. From a teacher's point of view, using VCSs increases the possibility of monitoring how students make progress with their assignments and eases the feedback process. The teacher could, for instance, include corrections and suggestions directly into the students' program code [1]. More generally, educators acknowledge that the use of VCSs relates to effective team work and that it is a crucial skill to be taught to prepare a competent workforce for present-day distributed workplaces [2].

An educational concern of interest to us is how students actually use VCSs. This has been previously studied by Mierle et al. [3] who investigated VCS usage patterns in a second-year course, hoping to find a correlation between an effective use of VCSs and study success. No clear patterns could be identified in the data which the authors attributed to the fact that beginner students climb their learning curve at different rates; see [3,4]. These authors call for more research on VCS usage patterns in particular in upper-year courses [4], which motivates the present study.

We have collected data about students' use of the distributed version control system (Git) from three different courses: *Introduction to Software Engineering* (second-year bachelor), *Functional Programming* (master's level), and *Service oriented architectures and cloud computing for developers* (master's level). A hypothesis arising from teacher observations during these courses is that students use VCS principally as a submission system rather than what it is intended to be. By this we mean that

- students commit at the end of the class sessions or right before the deadline, or there is only one commit per week/task,
- only one group member commits everything,
- students do not consider what file types to commit,
- overall, with no specific training, student do not use VCS efficiently.

We study these issues quantitatively exploring version control commit frequencies, commit sizes and the activity of individual students. Our specific research interest is the potential usage patterns identifiable in the commit log data of Git repositories.

1 Version control systems in education

Clifton et al. [5] summarize that in educational settings VCSs have been adopted to enable more realistic software development experiences for students [6], as a tool to monitor or visualize team and individual contributions [7], and for non-code artifacts such as creative writing [8]. Clifton et al. themselves, as well as many others, use a VCS for course management purposes. Further, some authors regard VCSs as a valuable tool to monitor and understand *how* students develop code [9]. Unsurprisingly, one of the most usual educational targets appears to

be courses with project work where VCSs both foster team work and facilitate course management tasks such as assessment and grading [10]. Milentijevic et al. [11] go as far as to propose a generalized model for the adoption of VCSs as support in a variety of project-based learning scenarios. All in all, we find that there is a general consensus of the benefits of VCSs as an integral part of computing curricula, one key argument being that they measure up to the requirements of globally distributed workplaces [2].

There are also challenges in the educational use of VCSs. Reid and Wilson [4], who used the CVS system, report on the confusion in judging which of the students' assignment versions was the final one. Glassy [9] found that students tend to put off working on assignments for as long as possible, even though a VCS is proposed to them with the hope of iterative work processes. Issues of this kind relate to inefficient use of VCSs. Furthermore, Reid and Wilson [4] noticed that some students mixed the functionalities of the CVS check out and update commands, and that also teaching assistants encountered problems if they had not properly familiarized themselves with the tool. These issues were considered to be due to a lack of a mental model of the VCS system used. Yet another challenge Reid and Wilson [4] raise is increased teacher workload when repositories are initiated and managed by teachers. In a more recent study, Xu [10] points out that there can be a long and rough learning curve before students feel comfortable using Git. Accordingly, Milentijevic et al. [11], who used CVS, report that students find a VCS to be a useful tool after they are sufficiently familiar with it. In the paper by Glassy [9] and Xu [10], the value of informative commit log messages is raised as a topic to be emphasized to the students. Rocco and Lloyd [12] in turn observed that some student have difficulties in understanding what constitutes "a significant change" to be committed.

It is much more difficult to find systematic empirical studies on issues such as how frequently students make commits and how they share the work. Rocco and Lloyd [12] found in their data that over 80% of a CS1 course population could adopt an iterative work process with the Mercurial system (50.0% did 7–21 commits and 33.3% more than 21 commits). On another course the authors defined a minimum commit frequency for one assignment and no requirements for the assignment that followed. With the first assignment, 75% of the students obtained a reasonable commit frequency, while with the latter this was 81%, altogether indicating that informing students of proper VCS usage can have a positive effect on their work processes. The authors note that not only were the students able to grasp the basics of the VCS (Mercurial), but they tended to continue to take advantage of the tool later on.

The present study focusing on the students' usage patterns with the Git system in both a second-year course and master's level courses complements the studies such as the ones by Rocco et al. [12] and Mierle et al. & Reid and Wilson [3, 4].

2 The courses

Introduction to Software Engineering (SE) is a 3-credit course consisting of lectures, a course assignment, and an end-of-course exam. The lectures introduce students to the basic concepts of software engineering, while the mandatory course assignment is the preparation of a project plan. The assignment is done in small groups and consists of four larger phases that need to be accepted by the lecturer. Mandatory supervision sessions on version control were arranged at the beginning of the course in order to encourage all the students to use the distributed version control system Git for the group assignment. The course had altogether 72 students in 33 groups (2.18 ± 0.76 students per group).

Functional programming (FP) is a 6-credit course implemented without traditional lectures and exams. The course is run in week-long cycles such that each week a new set of exercises is announced for the students. Students work in small groups and all of their study time is devoted to programming the weekly exercises. Two contact sessions are held each week. The first one is devoted to supporting the students' work and answering their questions. During the second weekly contact session there is a review of the student-written code. Overall, the course emphasizes self-direction on the part of students, similar to recently discussed course models such as the flipped classroom; see more details in [13–15]. Git was proposed for students as their primary group work tool and all of the exercises had to be returned via it. Thirty-six students were active in the course divided over 13 groups. (2.77 ± 0.89 students per group)

The last master's level courses studied, *Service oriented architectures and cloud computing for developers* (SOA&CC), introduces students to the use of digital services and the concept of cloud computing. A format similar to the FP course is used during the first (5 credits) part of the course. During that part of the course students undertake independent group work on a set of assignments each week. Two weekly sessions are arranged for the group work and one mandatory contact session focusing on reflective program review is arranged at the end of each week. An analysis of how the course model used in this course attempts to motivate students can be found in [16]. During the course Git is not only used as a version control system; it is also used as a tool to deploy code to Platform as a Service (PaaS) providers. Nine groups of students were formed with altogether 36 students (4 ± 0.82 students per group).

All three courses utilize the Faculty's *YouSource*¹ system. Similar to staff members, students can use their university credentials to log in to this system and create projects and Git repositories to manage collaborative work. The projects and repositories can be defined to be either private or public and collaborators can be added to them with a variety of permissions. This system has been in use at the department since mid-2010 and has been used in many courses and research projects.

It should be noted that in the remainder of this paper we are specifically concerned with the Git version control system, which belongs to the third gen-

¹ <https://yousource.it.jyu.fi/>

eration of version control systems (DVCSs). Students are free in their choice of environment for interacting with the version control system. Students can for instance use the *git* command line tool, tools with a graphical user interface, or tools included in their integrated development environments.

3 Data analysis

The Git repositories which students or course teachers created for the respective courses on the above-mentioned *YouSource* system are the source of all data analysis in this paper. One limitation of this data source is that we cannot see any data related to branches which a student did not push to the central Git server. However, if work of one of these so called local branches got merged into a branch which is synchronized with the central Git server, we are able to see its history as well. Further, this limitation is of minor importance since we are mainly interested in how students use the version control system in group work settings. Another limitation, which is inherent to the Git DVCS, is that we cannot know for sure whether time stamps on commits are truthful. It is technically possible to tamper with the date of the commits, but since there is no benefit for students to do so, we make the assumption that the time stamps are correct.

To study our research hypothesis, we will perform five different analyses, the first four of which are based on commits to the repository and the last one on the content of the repositories. For each commit we extracted the number of insertions and number of additions in accordance with the short status log of each commit². We added these two numbers together to form what we will call the number of changes of that commit. The tools used in the analysis have been developed by the authors of the paper and consume output produced by the diverse git commands.

For the first analysis we will, for each course, look at the commit activity over the whole course. To be concrete, we will visualize the commit activity by plotting the estimated probability density function of the total number of changes, i.e. for all students, over the span of the course. The density is estimated via the standard kernel density estimator, using a Gaussian kernel with bandwidth of 6 hours.[17] The height of the plot then shows the relative likelihood of a commit at a specific point in time.

The second analysis focuses on students' activity during the implementation sessions. This is done only for the FP and SOA&CC courses since the SE course does not have distinct sessions during which students get time to implement their work. We use a similar method as in the first part, but accumulate all commits that were made during the implementation sessions in the same plot. This plot shows when the students commit their code during the contact sessions. In the figure the far left of the x-axis represents the start of the session and the far right 15 minutes after the end. This is done in order to account for commits

² <http://www.kernel.org/pub/software/scm/git/docs/git-log.html>

right after the sessions. In this case we use a bandwidth which is one tenth of the total length of the session.

In the third part we perform an analysis of the commit messages in the different courses by classifying them in three categories : useful, trivial, and nonsense. A message is placed in the nonsense category if its content is not anyhow related to what is being committed. An example of this type of messages are these which contain only a couple of random letters, needed because the git system does not allow for empty commit messages. A trivial message is one which has no information beyond what is immediately visible from the commit meta-data. This category includes, for instance, a message consisting of a list of changed files or one saying that a given commit is a merge of two branches. All other commits are classified as useful. It should be noted that being in the useful class does not directly imply that the message is of high quality. It only means that the message is not trivial or nonsense. The classification was done manually by the respective teachers of the courses. We do not try to make a comparison between the courses, because the bias caused by having different raters is difficult to estimate.

In the fourth part we try to measure whether the version control system is used equally among the students in the group. If the system would be used by all students in a group, we would expect that the most active student in a group of n students performs $(1/n) * 100\%$ of the commits. To represent this number for all groups in the different courses we first find the students with the highest number of commits in their respective groups. Then we calculate their individual share in the total number of commits of their group. We then create an overview of the obtained percentages where we show different graphs for different group sizes since comparison among unequal group sizes would lead to biased results. It only makes sense to measure this for groups with more than one person. The SE course had a few single-person groups, hence only 28 groups from that course are included.

For the fifth and last part we investigate the types of files which students put under version control. First, teachers of each of the courses listed the file types and limits which they would expect a normal repository to contain. We started out from the files included in the HEAD of the master branch. For the FP and SOA&CC course we determined the type of each file using the BSD *file*³ command. The SE repositories required a manual analysis to decide the type of the files because the *file* command is unable to distinguish between the file types in use in the course. Then we counted the number of files of each file type. Then for each count, we compared it to the number of files expected by the teacher and any surplus was counted as garbage. The final number which we calculated for each group is the fraction of garbage in the total number of files.

³ <http://www.openbsd.org/cgi-bin/man.cgi?query=file>

4 Results

This section describes the results of our analyses. The first subsection shows the results for the analysis of the student activity during the whole course. In the second subsection, we focus on the implementation sessions only. The results of the analysis of the commit messages is shown in subsection three. Then we consider the activity distribution among students in the fourth subsection. Lastly, we look at the types of files which students submit to the version control system.

4.1 Commit activity over the whole course

The student activity in the SE, FP and SOA&CC courses is presented in the Figures 1, 2, and 3, respectively. In the SE course, we draw thick vertical lines to indicate the end of each of the four phases of the course assignment. As can be seen in that figure, there seems to be no correspondence between these deadlines and the student activity. In this course where VCS training was provided, students appeared to commit rather evenly throughout the course. We attribute the activity spike at the start of this course to the students trying out and getting familiar with the version control system at the point in time of the training sessions.

In the graphs of the FP and the SOA&CC courses (figure 2 and 3), we indicated with thin vertical lines the sessions during which students get time in the classroom to work on the assignments. The thicker vertical lines indicate deadlines for the weekly assignments. The dates of the sessions are displayed on the x-axis in a month/day format. In contrast to the SE course, these graphs show a closer correlation between student activity and the implementation sessions and the deadlines. The graphs suggest that most of the work was committed during the contact sessions, which again suggests that students bring their work to the sessions to be committed there. This prompts us to study the student behaviour during the sessions separately in the next section. It is also clearly visible that the students have a very low activity during the weekends.

4.2 Commit activity during the sessions

In the FP and SOA&CC courses students were more active during sessions than at other times. The graphs in figures 4 and 5 show the students activity during the sessions and 15 minutes after the session. We normalized the duration of the session (90 minutes) and the 15 minutes overtime between zero and one. Interestingly, we notice a similar behavior in both courses. There seem to be three periods of higher activity. The first moment of higher activity is in the beginning of the session after about 10 minutes. The second one, which last longer, is between 20 and 40 minutes after the start of the session and lastly, the activity peaks shortly after the session.

The first period of activity is most likely because individual students have been implementing parts at home. These students then decide to commit only

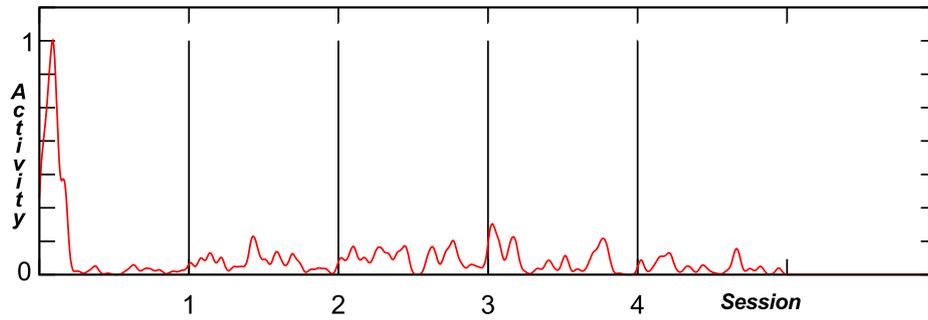


Fig. 1. Commit activity during the SE course

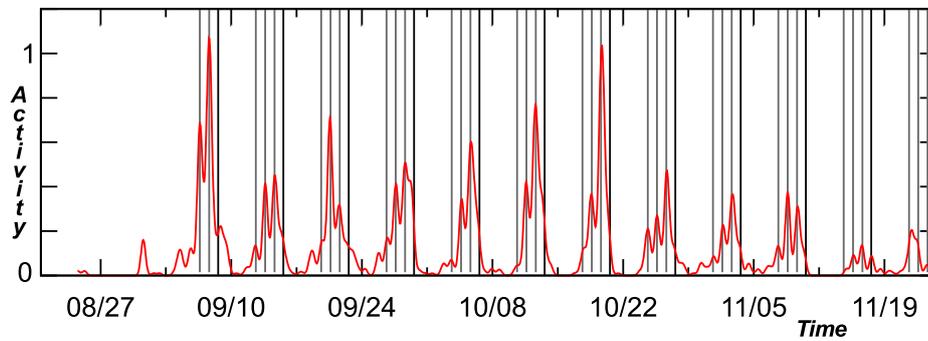


Fig. 2. Commit activity during the FP course

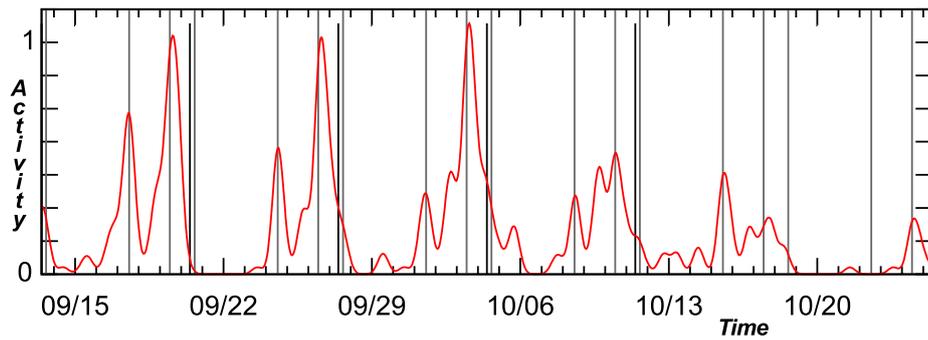


Fig. 3. Commit activity during the SOA&CC course

after receiving consent from other group members. This is an indication that students do not know how to use the version control system efficiently, as in principle they could have used a separate branch for their local development and merged their changes to their shared version of the exercises. Also, speculating based on student dialogue, some students might have feared 'losing face' by making their preliminary versions visible to others, including the teacher.

During the second period of activity students are using the system as they are supposed to, committing changes regularly. Then the activity drops for quite some time before reviving shortly after the session. We attribute this last peak to those groups who have been working during the whole session without committing many changes. At the end of the session they want to store their work for later continuation and decide to put all their work in the system.

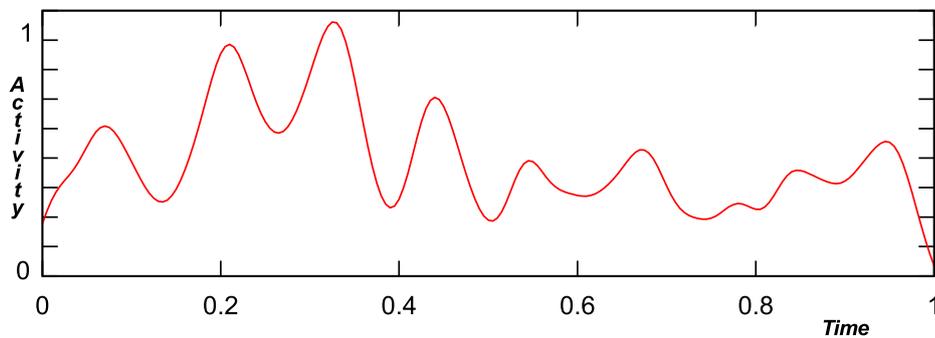


Fig. 4. Commit activity during the sessions of the FP course

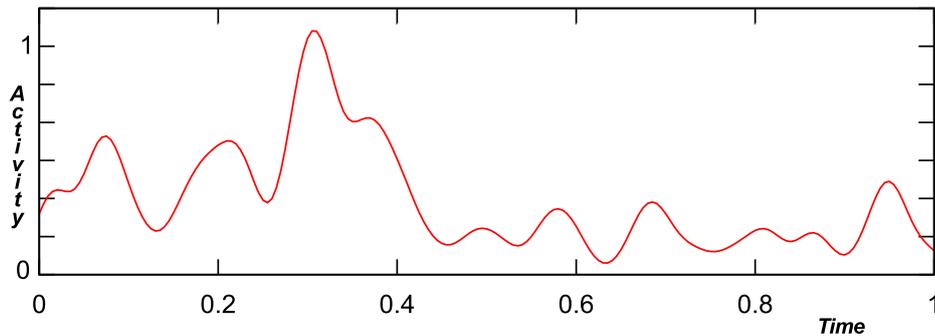


Fig. 5. Commit activity during the sessions of the SOA&CC course

4.3 Commit message analysis

The classification of the commit messages was performed for the SOA&CC and FP courses and yielded the results shown in table 2.

Table 1. Categorization of the commit messages per course

	useful	trivial	nonsense
SE	996 (67%)	430 (29%)	59 (4%)
FP	1422 (78%)	276 (15%)	129 (7%)
SOA&CC	289 (74%)	65 (17%)	37 (9%)

Table 2. Categorization of the commit messages per course

In an ideal repository we would not find any trivial and nonsense messages. What we see from the table however is that there is a significant amount of these types of messages.

It is not visible from the table, but the teachers classifying the messages shared the opinion that the messages in the useful category were not all that descriptive. Some commit messages could be regarded as ‘locally sensible’, meaning that they could be useful for communication during a short time span, but offer not much for later inspection. Many of the commit messages are clear indicators that the students regard the system as an answer submission system. Examples include “Answer for week 12” and “exercise 4a”. We also noticed some messages related to problems in using the git system. The amount was however not as significant as the teacher had expected. It is also observed that the quality of the messages is depending on the group, indicating that some groups use the messages for communicating, while others do not.

4.4 Differences in student activity

To show the differences in activity among students we assembled the charts in figure 6. The figure contains one pie chart for each course and each group size or none if there are no groups of the given size in the course. Each pie chart illustrates the fraction of the groups which have a given percentage of commits for their most active committer. The last column shows the expected fraction, i.e. the chart which would be obtained if all students in the groups do an equal number of commits.

What we see from the charts is that the most active committer in a group, most of the times, commits significantly more as the expected percentage. Put another way, the most active committer in each group is very often much more active as the average which one might expect. This can be due to that student having a dominating role in the group. In the FP and SOA&CC course we tried to mitigate this effect by grouping students according to their skill level.[13, 15, 16] We however think that the main differences are caused by a different level

of familiarity with the version control system between the group members. The person with the most experience will commit more frequently or is given the task of submitting the work of others to the system.

4.5 Which files did the students commit to VCS?

The presence of files which do not belong in the version control system, such as executable programs, temporary compilation files and copied documentation, suggests that the students used the VCS as plain file storage. Figure 7 shows the fraction of the groups with a given percentage of redundant files in their final repository. We see a big difference between the figures for the respective courses.

In the graph for the SE course we see that most groups did not include many compiled files in their repositories, as was explicitly instructed in the course.

During the functional programming course, students can often test their code without actually compiling it, which could explain the low amount of garbage in the repositories. The garbage that is committed consists entirely of compiled binaries and other compiler generated files.

During the SOA&CC course many students use integrated development environments (IDE) which do the compilation automatically for the user. It seems like many students have included all files which the IDE produced to the version control system.

It seems that if students are not made aware of the fact that they should not include this kind of files to the VCS, they tend to include everything that happens to be present in their local directory. We should do further research to see whether this behavior changes if students are made aware of the best practices.

Conclusion

In this article, we focused on students' usage patterns in advanced courses related to programming while using the distributed version control system Git. We first looked at when students commit their work during the course and in more detail at their committing pattern during the implementation sessions. We concluded that most students commit changes regularly during the implementation sessions, but do not commit changes of work which they have been doing before the session itself. Some groups commit rarely during the session and make a big commit at the end of the session. We did some effort in classifying commit messages and noticed that students do often write messages which are either trivial or even sheer nonsense. Further, we looked at how the usage of the system is divided inside groups and found out that the activity of the most active user in a group is significantly higher than what would be expected if each group member would use the system equally much. As the last part of our analysis we considered the types of files which students put under version control. We concluded that if students are not told explicitly that they should not include certain types of files, they will just do so.

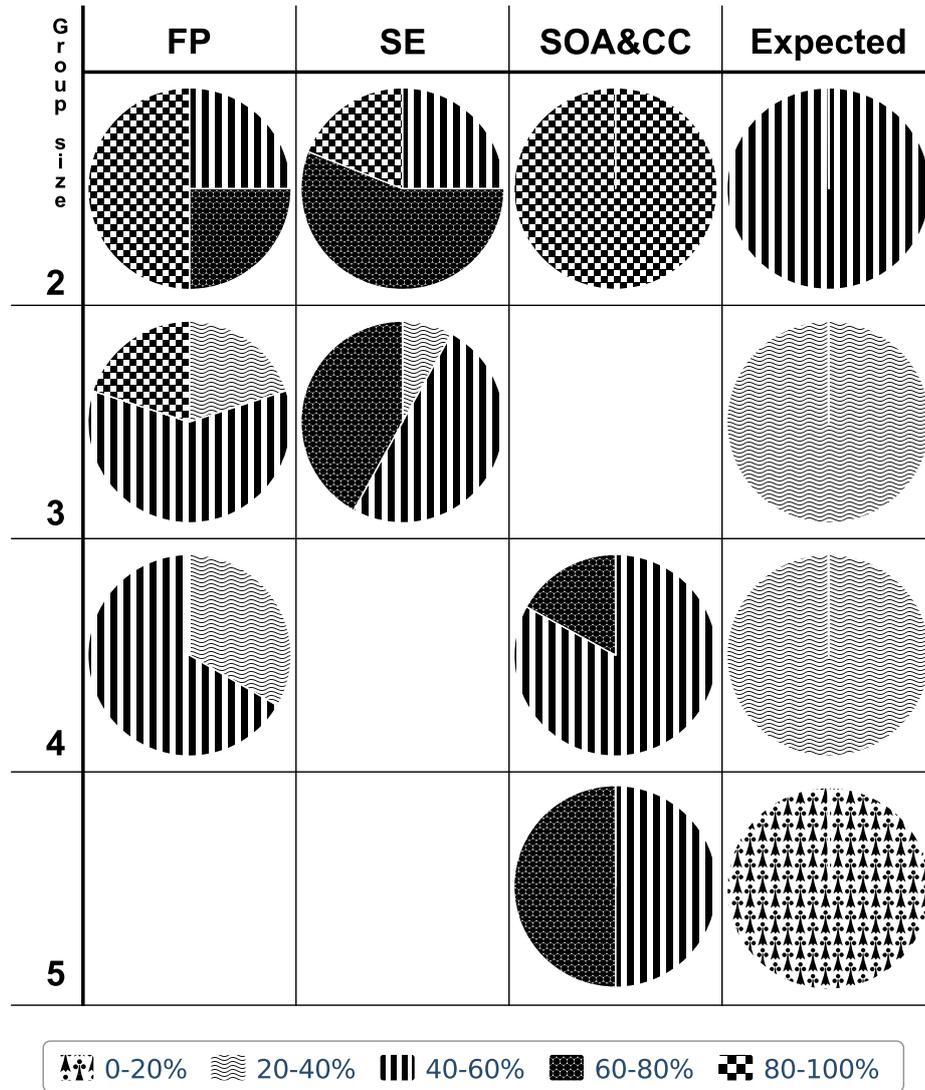


Fig. 6. Fractions of the groups with a given percentage of commits performed by its most active student

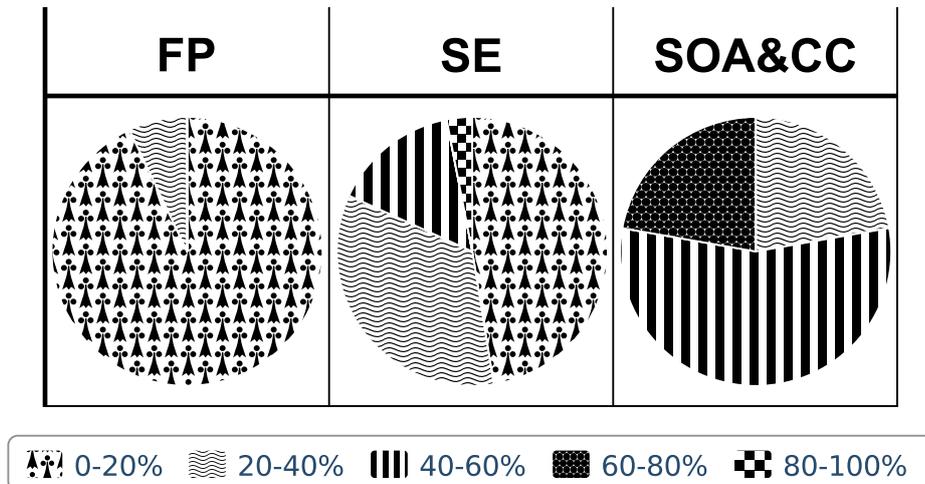


Fig. 7. Fractions of the groups with a given percentage of 'non-versionable' files in their repositories

With regard to the hypothesis put forward in the introduction, our findings suggest that using a VCS as a submission management tool may result in students adopting the tool as “just a required manner to return the assignments” — instead of a professional tool by which collaborative and distributed work is managed. Indications for this were that across all three courses studied the version control system was not too evenly used by the team members and the fact that the quality of the commit messages was quite low. While VCSs are pronounced as useful course management tools in the literature, we would like to note that professional use of VCSs requires support and demonstration of their usefulness.

In our future undertakings, we could make a distinction between submissions returns and use of VCS, and add VCS training to the beginnings of the courses. Further, the existing classroom setup where there is sometimes only a single computer for the whole group could be replaced with settings where each student would use a separate computer. Performing these practical changes could reveal whether a more intense and shared use of a VCS can be prompted among the students. Promisingly, in our second-year course, training sessions were provided and there were no observable commit peaks near the deadlines of assignment phases but a rather constant commit curve. Further research could also point out how effective the students can use the system and how the organization of the group work influences the use of the system. It might be that some students know very well how to use the system, but do not see any reason to use their skills up to a full extent in the given settings.

Acknowledgments: We would like to thank the department of Mathematical Information Technology of the University of Jyväskylä for both financial

and material support, without which this research would not have been possible. We would also like to thank the reviewers for their useful corrections and suggestions, which greatly improved the quality of this paper.

References

1. Laadan, O., Nieh, J., Viennot, N.: Teaching operating systems using virtual appliances and distributed version control. In: Proceedings of the 41st ACM technical symposium on Computer science education. SIGCSE '10, New York, NY, ACM 480–484 (2010)
2. Meneely, A., Williams, L.: On preparing students for distributed software development with a synchronous, collaborative development platform. In: Proceedings of the 40th ACM technical symposium on Computer science education. SIGCSE '09, New York, NY, ACM 529–533 (2009)
3. Mierle, K.B., Roweis, S.T., Wilson, G.V.: CVS data extraction and analysis: A case study. technical report utml tr 2004-002. Technical report (2004)
4. Reid, K.L., Wilson, G.V.: Learning by doing: Introducing version control as a way to manage student assignments. In: Proceedings of the 36th SIGCSE technical symposium on Computer science education. SIGCSE '05, New York, NY, ACM 272–276 (2005)
5. Clifton, C., Kaczmarczyk, L.C., Mrozek, M.: Subverting the fundamentals sequence: Using version control to enhance course management. SIGCSE Bull. **39**(1) 86–90 (March 2007)
6. Hartness, K.T.N.: Eclipse and CVS for group projects. J. Comput. Sci. Coll. **21**(4) 217–222 (April 2006)
7. Liu, Y., Stroulia, E., Wong, K., German, D.: Using CVS historical information to understand how students develop software. In: MRS 2004: International Workshop on Mining Software Repositories. (2004)
8. Lee, B.G., Chang, K.H., Narayanan, N.H.: An integrated approach to version control management in computer supported collaborative writing. In: Proceedings of the 36th annual Southeast regional conference. ACM-SE 36, New York, NY, ACM 34–43 (1998)
9. Glassy, L.: Using version control to observe student software development processes. J. Comput. Sci. Coll. **21**(3) 99–106 (February 2006)
10. Xu, Z.: Using git to manage capstone software projects. 159–164 (2012)
11. Milentijevic, I., Ciric, V., Vojinovic, O.: Version control in project-based learning. Computers & Education **50**(4) 1331–1338 (2008)
12. Rocco, D., Lloyd, W.: Distributed version control in the classroom. In: Proceedings of the 42nd ACM technical symposium on Computer science education. SIGCSE '11, New York, NY, ACM 637–642 (2011)
13. Tirronen, V., Isomöttönen, V.: Making teaching of programming learning-oriented and learner-directed. In: Proceedings of the 11th Koli Calling International Conference on Computing Education Research. Koli Calling '11, New York, NY, ACM 60–65 (2011)
14. Tirronen, V., Isomöttönen, V.: On the design of effective learning materials for supporting self-directed learning of programming. In: Proceedings of the 12th Koli Calling International Conference on Computing Education Research. Koli Calling '12, New York, NY, ACM 74–82 (2012)

15. Isomöttönen, V., Tirronen, V.: Teaching programming by emphasizing self-direction: How did students react to active role required of them? *ACM Transactions on Computing Education Research* (accepted)
16. Isomöttönen, V., Tirronen, V., Cochez, M.: Issues with a course that emphasizes self-direction, submitted. (2013)
17. Parzen, E.: On estimation of a probability density function and mode. *The annals of mathematical statistics* **33**(3) 1065–1076 (1962)

Comparative Analysis of Learning in Three-Subjective Didactic Model

Aleksander Spivakovskiy¹, Lyubov Petukhova¹, Evgeniya Spivakovska¹,
Vera Kotkova¹ and Hennadiy Kravtsov¹

¹Kherson State University, 27, 40 RokivZhovtnya St., Kherson, 73000, Ukraine

{spivakovsky, petuhova, spivakovska, veras, kgm}@ksu.ks.ua

Abstract. The article theoretically shows transformation of modern didactic model into three-subjective (Student - Teacher - Information and communication pedagogical environment). Active components of new subject, which are the most evident in a learning process, are analyzed in the article. The requirement block of information and communication environment as a subject of the educational process is described. The comparative characteristic of the main components of traditional and innovative teaching systems is presented in the article. The authors have made a comparative description of the main forms of university studies in different didactic models: object-subject, subject-subject and three-subject training. The measurement of cogency of each of these three study subjects and their significance in the process of major educational operations (collection, processing, storage, transmission) in various forms of training: lectures, practical classes and individual work were presented.

Keywords. Didactics, information society, information and communication pedagogical environment, three-subjective didactics, forms of training organization at university

Key terms. KnowledgeEvolution, KnowledgeManagementMethodology, Didactics, KnowledgeManagementProcess, ICTInfrastructure

1 Introduction

Education is an institute of social experience transmission and human socialization in society. Naturally it depends on the level of social development and labor market needs.

Modern university education is in crisis, according to UNESCO specialists' definition, it helpless and failure of modern education may bring to global problems of humanity. These are irregular development of different countries in the context of globalization, education inactivity caused by the relative conservatism of human resources brings to constantly fast-moving knowledge renewal.

Weak sides of university education are the following students' training instead of

general cultural development, low professional motivation and responsibility, strict regulation of students' activities, provided graduates' inactivity, not much attention to the levels of training, etc. According to this fact, it's said about global educational crisis, the paradigm shift in pedagogical thinking [1].

We are going to trace the change of professional education at different stages of society's development to overcome the crisis of modern university education.

Down the ages human language existed primarily in form of sound speech. Its main limitation was space-time limit: spoken word spread out the territory limited by physical laws of sound and in form of material reality actually existed only while pronouncing, straight after that passing into the history and vanishes in it. The era of word was characterized by a certain lack of knowledge acquirement and the Institute of transmission, as the main source of the word transmission process from one generation to another appeared to be a man.

The increase of information amount became the background for writing nascence as it was difficult to keep the information in mind without losing its content. Writing unlike the sound speech turned to be the technology of knowledge transfer.

The invention of writing (i.e. the possibility of fixing speech using a specially developed system of graphic symbols) allowed to transmit voice information to an unlimited distance and extremely broadened its existence in time. Beyond dispute, the appearance of writing created new additional conditions and opportunities to realize a potential of human culture. But at the same time, writing leads to limitation and narrowing of informational content of speech. An issue is that writing is a sign system and it is shown as a representative of the signified so it reproduces only a half of properties and meanings of what it means. In this case, word transfers only a part of the properties and meanings contained in the "live" speech. Thus, written language is actually completely lost in the so-called prosodic information contained in the "live" speech that sounds. The case is that the graphical definition is losing information that is expressed and transmitted in direct speech by means of phonetics which plays a divisionary role.

Year 1450 AD (500 years ago) is marked by the appearance of new information technology, the third one in a row. Only then printing technology appeared which we consider a knowledge distribution technology. We call this phase an era of books. Definitely the appearance of books allowed creation of an effective and mass education system, to organize public libraries, to ensure the development of universities. The appearance of books as a mean of transmitting knowledge, promoted the human-kind's achievement of those heights which it has now.

An important consequence of definite social development turned to be an understanding of purposeful activity of social skills transfer from one generation to another as a connection between two organized activities – teaching and learning, their concrete reflection in the learning process. Humanity cumulative experience of the learning process has found expression in didactics, one of the pedagogy's section that examines general theory of education and training. It is believed that the term was introduced by German pedagogue Ratko in his lectures "Rahitiy's summary of didactics and art education", meaning a scientific discipline which studies theory and practice of teaching [2].

Efforts to make an educational process intelligently organized and purposeful are presented in many Jan Komenskiy's works, especially in "Great didactics," which covers almost all the issues that present the subject of modern pedagogy. Jan Komenskiy was the first one who developed didactics as a system of scientific knowledge, giving a reasoned exposition of principles and rules for children's education. He examined the most important questions of the learning theory: educational content, teaching visualization principles, sequence of education, organization of class-and-lesson system, etc.

Object-subject teaching relations between teachers and students of that time were the most prolonged in pedagogy. The subject is the teacher who works actively to educate students as his objects of influence through *informative-educational environment* (word, book, equipment). Schematically such didactic relationship is depicted in Fig.1.

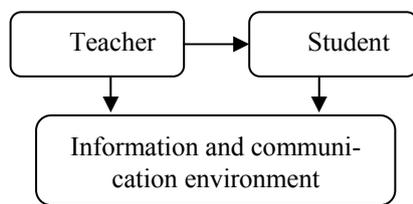


Fig. 1. Schematic model of the object-subject relations

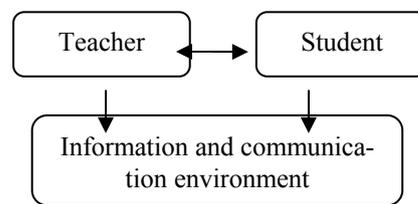


Fig. 2. Schematic model of the subject-subject relations

Gradual development of public experience of mankind has increased to such an extent that a person using only natural abilities was not capable to learn and operate with informational resources. As the result, the person begins to use technological tools to optimize working process with information. According to it, labor market no longer corresponds to specialists' "conveyor" training, as received mental vocabulary knowledge quickly becomes obsolete, an employee must make a "decision" in unusual situations. Naturally, a student becomes an equal subject of the educational process. The transformation of an object into a subject in the educational process is the result of the democratization of education, dissemination of differentiation and individualization of education. Schematically, the new relationship is depicted in Fig. 2.

Intensification of information processes, introduced into science, economics, production, requires the development of new models of education, a variety of information and communication environments in which people could reveal their creativity fully, develop skills and cultivate a necessity for self-improvement and responsibility for their education and development.

The traditional paradigm considered education as training of younger generation to work and life by consuming material valuables created in other areas. The new paradigm foresees independent values in education.

2 Innovative methodical system

The purpose of creating a new education paradigm is to provide conditions for education, training and development for independent, smart person to satisfy the requirements on a market economy, capable to improve continuously his own level of knowledge and culture, integrated into the global informative space.

Thus, today, we are talking about innovative methodical system which unlike traditional one, corresponds to professional education demands in an informative society. The comparison of the main components of these two systems is presented in Table 1 [3].

Table 1. Comparative characterization of traditional and innovative teaching systems.

Name of component	Traditional methodical system	Innovative methodological system
Learning objectives	Seizure and adoption of educational material. Provide students with knowledge, skills and practice.	Provide students with knowledge, skills and practice. Creation of modern information and communication teaching environment. Purposeful development of creative self-sufficing person. Formation of professional competence, leadership skills, ability to work in group.
Principles of learning	The scientific character principle. The principle of systematicity and consistency. The principle of visibility. The principle of studying direction in accordance with issues of education, training and development.	The principle of the activity learning environment. The principle of organic unity between the changing requirements at labor market and conserved features of the educational system. The principle of necessity for continual self-study.
Contents of training	Classical learning, technocratic one.	An integrated approach to fundamental and applied activity aspects of a specialist-to-be.
Study methods	Reproductive, explanatory, illustrative.	Problem-search, research.
Study means	Visual tools. The teacher's word - for knowledge transfer, books, movies, tape, training devices, pictures, maps, tables, machines, devices, models, collections, tools, and historical schemes, charts,	Facilities. Information and communication technologies. Hypertext, multimedia training materials. Databases for educational purposes. Networking means for videoconferencing and video lecture. An effective system of

Name of component	Traditional methodical system	Innovative methodological system
	diagrams, etc. Technology. Video-recordings, radio and television, filmstrips, slides, transparencies, projectors, televisions.	monitoring training activities. Remote devices for self-work. Computer testing in on-and off-line modes.
Study forms	Lectures, seminars and practical lessons.	Dispute, seminars, conferences, "round table", symposium, debates, colloquium, distance learning, teaching and business games, role-play game.
Control forms	External process operations control within strictly defined rules is dominated. A teacher assessment result (flow, final control) is dominated. Lack of balance between control and self-control. Lack of effective control for individual learning methods of each student.	Strict current control of individual learning of each student by means of testing in on-and off-line mode. Rating control knowledge. Creating an effective environment according to Jean Piaget for easy convenient self-organization that motivates students in learning activities.

In addition, society today has faced the phenomena which require answers:

1. Teacher has lost the monopoly on knowledge;
2. Students have unlimited access to information resources;
3. The phenomenon of "red shift" in expanding informative and communicative space;
4. Availability of qualitatively and quantitatively different ICT competencies of young and older generations.

For that matter, an educational paradigm transforms, which is characterized by the following principles:

- Globalization of knowledge, free access to educational resources
- Integration of learning resources
- Organization of global educational audiences
- WEB-multimedia presentation of educational resources
- Multilingual educational space
- Asynchrony of modern models for learning management
- Harmonization of social and educational environment
- Formation of social identity of information system
- Divergence in the implementation of their own educational way

Thus, an evolution of modern education, information studies, mass computerization of educational establishments, constant upgrade of hardware, and development of

computer networks, expanding of personal computerization of society, increasing of software products designed for use in an educational process – these are conditions that create new *information and communication pedagogical environment* (ICPE). This environment constantly and aggressively increases student’s motivation to consume content that circulates in it, creating a new didactic model – *three-subjective relations*, which include three subjects of study - students, teachers and an environment.

However, is it legitimate to consider ICPE a possessing equal rights subject for learning along with a teacher and a student?

3 Model of three-subjective relations

Consideration of information and communication teaching environment as a subject, in our opinion, is possible because its components are not only technology but human resources as well, which continuously update them at the constantly growing speed. In this sense, it is necessary to point out an existing qualitatively new learning environment as opposed to which one that was 15-20 years ago. The question deals with the obtaining of today's educational environment the status of an equal partner. Sir Ken Robinson in *The Third Teacher* (2010) says, “The physical environment of the building is critically important in terms of curriculum” [4].

Within this approach, we implement an important target triangle: a natural integration of teaching, research and labor market needs. After all, ignoring the environment as a subject of education, we will prepare specialists for inadequate reality.

The inevitability of the transition of the education system to consider three-subjective relationship is reflected in the following three stages of didactic changes:

Stage I – the subject-object instruction (a teacher provides students with knowledge). Characterized by one-dimensional linear model, the volume of processed data – megabytes;

Stage II – subject-subject didactics (a teacher and a student are equal competent training partners). Characterized by two-dimensional polylinear model, the volume of processed data-gigabytes;

Stage III – three-subjective pedagogy (a teacher – a student – ICPE). The interaction of all subjects of the learning process (a teacher – a student – ICPE) obeys to the common goal which is formation of a competitive specialist and is characterized by a three-dimensional nonlinear model, the volume of processed data – terabytes.

Thus, we have the right to talk about three-subjective didactics as one of the areas of pedagogical science of the most general regularities, principles and means for organization of studying, providing a firm and conscious assimilation of knowledge and skills within peer relations pupil (student), teacher (teacher) and information and communication teaching environment.

It is important to underline that in this process, status and general condition of those who learn and teach and ICPE are constantly changing. In this context, we understand these learning activities with assimilation of knowledge and skills, and teaching - the knowledge message or source of knowledge to students, as well as in-

struction on ways and methods of work, coordinating training activities, particularly organization of active forms (discussion, round table, project activities, etc.) and monitoring of students mastering knowledge, skills and experience obtaining. Unlike traditional views, we consider, that it's necessary to introduce the one who teaches into the learning process, the changes that are ICPE (for example, by means of publishing of educational materials in the Internet). We have to mention, that new innovative forms of teaching activity are connected remotely or, as they say, distance management software training activities, both in time and space.

Within this definition naturally occurring three-subject relations, which we understand as the continuous and constant (both in space and time) interactions between students, teachers and information and communication pedagogical environment directed for satisfaction of students educational needs (Fig. 3).

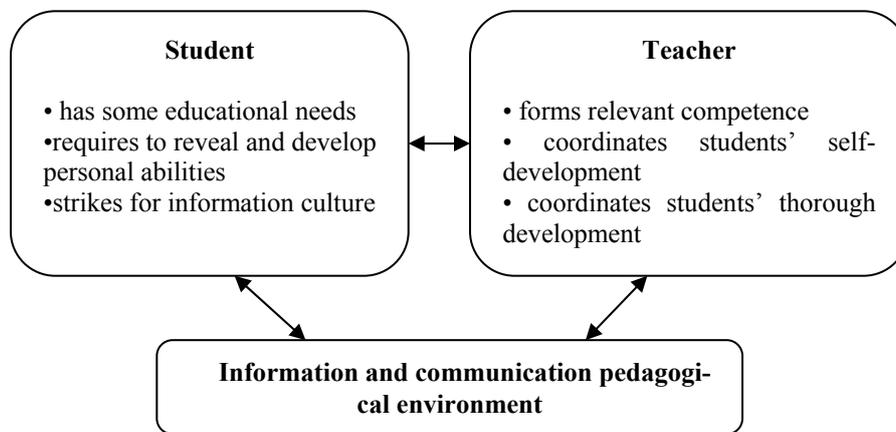


Fig. 3. Schematic model of three-subjective relations

As we are examining an environment as a separate element, it's important to mention its operation characteristics, which are the most evident in the learning process:

- environment constantly and more aggressively increases motivation of the younger generation for content use that circulates in it
- environment provides access to resources at any convenient time
- environment has comfortable, flexible, friendly, intelligent service, that helps people to find informational resources, data or knowledge they need
- environment is not a negative emotional one, it corresponds people's demands as much as it is needed
- environment permanently is filled up with information, data, knowledge with a constantly increasing speed
- environment offers an opportunity to organize practically free, time convenient contacts between any number of people to provide suitable and flexible information exchange (in any form) between them

- environment, step by step, standardize, and then integrates the functionality of all previous, so-called traditional ways of receiving, storing, processing and presenting of the required information, data and knowledge to mankind
- environment undertakes more and more routine operations connected with humans operating activities (which is one of the greatest challenges for humans to expect in the future - "the more commissions - the more responsibility - the greater risks remain without resources")
- environment receives more and more control over the data, and operational mankind's activities [3]

Due to our three-subjective didactics, we can answer the above-listed vital questions connected to modern educational system:

- the teacher's role and place in the new didactic model
- correlation between virtual and visual forms of subjects' relationships in didactic system
- development of technological management providing rights for subjects of didactic system to login informational resources
- organization of modern control systems over learning activities
- assurance of the organic unity between changing requirements of labor market and conservative educational system potential
- organization of modern, and most importantly, systematic and constantly active system of re-training and professional qualification upgrade of teachers

Step by step, it is getting clear that technology that produces modern industry, today, not only affects the technology transfer of knowledge, but in fact, it determines qualitatively new forms of its organization for their mastering. At this stage, we can see the following problems:

1. Heterogeneity of distribution of computer and communication facilities
2. Huge differences in the process of training and constant re-training of staff, both academic and administrative ones
3. Inertia of education system
4. Constantly growing volume of technological renewal of learning environment that includes all the tools, both for teacher and learners
5. Imposing of different learning paradigms that make substantial confusion in the teachers' presentation of their new role in the process of knowledge transmission, development of abilities and skills
6. Stereotype of the philistine attitude to pedagogy in whole, as a descriptive section of human knowledge, in which every citizen is a knowledgeable expert
7. Absence of formal systems which describe different models of learning [3]

Active learning environment contains the following units, which are procedural, substantive and control. The environment begins to play a more important role and assumes some part of teacher's functions. There is no doubt that, certain requirements must be done in the process of setting up a proper learning environment, which will provide active learning environment. Working with the program, both a student and a teacher will be limited by a system of actions, which was laid out in the program,

that's why development of the system requirements of ICPE is very important. According to our research, information and communication learning environment can serve as the subject of the educational process if it meets the following group requirements:

1. Hardware requirements: multimedia computers in classrooms are networked with the obligatory access to the Internet resources. In addition, an important aspect creates opportunities to access educational electronic resources (Wi-Fi technology) for students in any convenient place, for example, library, dormitory, canteen etc.
2. Software requirements: software environment should resolve security issues (registration, personalization, delineation of access rights to get to resources), to be integrated (all educational components should be in its natural form), easy for exploitation, filling and modification, to provide opportunities for interaction, communication, monitoring for learning process, to contain an output mode out of the complicated situations (expert), to offer opportunities for distance learning (on- and off-line modes).
3. Academic requirements refer to methods of filling information and communication teaching environment.
4. Social demands. Special attention should be paid to a specified group of claims which, in our opinion, contains cultural, ethical and legal aspects, because users of information and communication pedagogical environment create some community. First of all, it is about the rules of communication in the network and use of the reworks of other authors.

Requirements to Human Resources. Construction of the educational process on the basis of information and communication technologies implies specialists- programmers and accordingly well-trained teachers.

ICPE correspondence to these requirements can be achieved by using management system of the quality of educational information resources [5].

Introduction of new subject of learning process naturally transforms existing elements of training, including forms of teaching at higher educational establishments. Today, educational resources are open and distance forms for studying are actively developing and integrating into traditional forms of teaching: lectures, workshops, laboratory classes, independent, individual work of students, forms of control. Let's try to analyze basic traditional forms of training organization in different didactic models (Table 2).

Table 2. Forms of learning in didactic models

Subject-object study	Subject-subject study	Three-subject study
Teacher		
The source of educational information is a teacher; students are forced to put down a limited amount of	A teacher presents difficult educational material, students selectively put down the information that is nec-	A teacher and students in the debate form discuss problematic issues due to free access to open lecture and other information

Subject-object study	Subject-subject study	Three-subject study
information, static visibility is used additionally.	essary for each personally, use additional sources, including the Internet. Dynamic visibility is dominant.	sources. Students write down the required information at will.
Practice		
Reproductive methods of teaching material development are used.	Part-search training methods prevail.	Search and creative methods are directed at forming experience of training materials, particularly under unusual circumstances.
Independent work		
It consists of lectures, practical exercises execution.	Studying unwrought amount of teaching material.	The main part of teaching material is studied individually.
Forms of control		
A control requires presence of a teacher who relates a student's knowledge with the volume of lectures material.	Students' readiness to use received knowledge in condition of life situation is also under control.	Monitoring can be conducted without teacher's presence, and the result – unconventional approach and creative thinking of students are estimated.

4 Measurement of the importance of training subjects

However, is ICPE a significant, important subject of learning in practice of University operation? The theoretical conjectures study was conducted at the Faculty of pre-school and primary education of Kherson State University in order to confirm or refute it. The research required a questionnaire of future primary education teachers, as they acquire an integrated system of philology, humanities, exact, natural and artistic sciences, which, in our opinion, reduce a risk of results' obtaining only from certain cycle of training. The main task of the questionnaire is to evaluate the significance of subjects of modern educational process, including ICPE.

Determining the validity of each of the three subjects of the educational process was made by expert evaluation method. 27 qualified experts (university teachers, graduate students, methodologists) joined the independent expert committee.

To define a point of evaluation for each subject Delphi method (for members of the expert committee conditions for an independent individual work were created) was used [6]. Maximum and minimum estimates depended on a number of subjects, in which, there are three. Thus, minimum score for one of three components – 1 point, an average score – 2 points and maximum – 3 points. Then, the statistical processing of the results, which were presented to experts for final approval, had been conducted. The cycle of expertise was repeated three times.

Below are the results of an independent expert committee (Table 3).

Table 3. Determination of cogency of training subjects (*V*)

Subjects of the educational process	Number of points			Σ	<i>V</i>
	1	2	3		
Teacher	12	8	7	49	0,30
Student	2	10	15	67	0,41
ICPE	12	11	4	46	0,29

According to the results of expert reviews cogency *V* (in fractions of a unit) for each of these three specified subjects, according to experts, is approximately the same, with a slight advantage "student" (0.11 larger compared to "ICPE" and 0.12 larger compared to the "teacher").

Results summarizing the data are shown in Fig. 4.

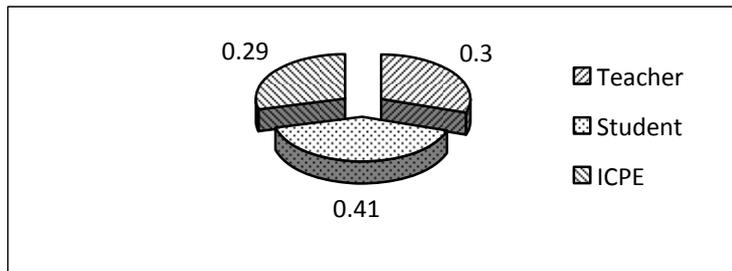


Fig. 4. The importance of the subjects of the educational process

214 students, as the most significant subject of didactic system according to experts' definition, were asked to rate on a 10-point scale the importance of the three subjects of the educational process: students, a teacher and ICPE – in the process of operating with information (collecting, processing, storing, transmission) in various forms of training organization: lectures, practical classes and independent work. To do this, students were asked to determine the importance of each component of didactic models: Student - Teacher - ICPE a five-point scale (1, ..., 5).

Results of the student's questionnaire are shown in Table 4.

Assessment E_{ij} ($i, j = 1,2,3$) for each *i*-component of the didactic system operations in terms of *j*-forms of training organization is given by (1):

$$E_{ij}=K_{ij1}+K_{ij2}+K_{ij3}+K_{ij4}, \tag{1}$$

where E_{ij} — total score in terms of transactions weight, K_{ijk} - *i*-score weighting components didactic system, *j*-teaching forms and *k*-rate transactions, %.

The overall assessment of V_i ($i = 1,2,3$) for each component of the didactic system is given by (2):

Table 4. The results of the student’s questionnaire

Components		Student		Teacher		ICPE	
Form of training organization	Indicators	points	%	points	%	points	%
		Lecture	Collecting	1182	5,5	2187	10,1
Processing	2151		10,0	2630	12,2	2411	11,2
Storing	2625		12,2	1455	6,7	2402	11,1
Transmission	830		3,9	2004	9,3	1355	6,3
Σ	6788			8276		6502	
<i>E</i>			31,6		38,3		30,1
Practice	Collecting	1674	7,7	2006	9,2	1235	5,7
	Processing	1885	8,7	3121	14,3	815	3,8
	Storing	1241	5,7	1663	7,6	2421	11,1
	Transmission	2178	10,0	1198	5,5	2322	10,7
	Σ	6978		7988		6793	
<i>E</i>			32,1		36,6		31,3
Independent work	Collecting	2214	9,7	2119	9,3	2033	8,9
	Processing	2366	10,4	2882	12,6	1938	8,5
	Storing	2154	9,4	2007	8,8	2013	8,8
	Transmission	994	4,4	384	1,7	1703	7,5
	Σ	7728		7392		7687	
<i>E</i>			33,9		32,4		33,7

$$V_i = (E_{i1} + E_{i2} + E_{i3}) / 3. \tag{2}$$

Let’s analyze the results.

It is generally known that, lecture – is the main form of teaching, prepared for the adoption of theoretical material. Table 4 shows that while gathering information during lectures (17.1%), the most significant entity of the educational process is a teacher (10.1%), 5.5% of operation is performed by a student, 1.5% – ICPE. It’s explained by identification of the content and material of lectures, in its selection, the main role is occupied by a teacher, but the lecture provides not passive acceptance of students’ knowledge but their active involvement into the learning process, preparation for lectures, which is provided with ICPE use. We should note that active cognitive activity of students during lectures is possible for basic training, which includes familiarization with the theme of the lecture and its plan, the main content of the theme for the tutorial, content repetition of the previous themes etc.

According to the survey results, information processing on the lecture (33.4%) subjects’ contribution is approximately the same: 12.2% - Teacher, 11.2% - ICPE, 10% - Student. This is because the teacher coordinates educational information processing, and an active entity involved in this process may be a student. ICPE activity due to a shift in emphasis onto the use of methods and means of processing students - from note-taking information material: full or theses synopsis for computer processing of the information received.

Storing educational information of the lecture (30%) between the subjects of the educational process was distributed: 6.7% - Teacher, 11.1% - ICPE, 12.2% - Student.

According to received questioning results, transfer of educational information of the lecture (19.5%) is implemented by a teacher (9.3%) and ICPE (6.3%), although students (3.9%) provide additional information, interesting facts and problematic issues. The task of the teacher, at this stage, is to transfer the adapted information disclosing a nature of scientific concepts, genesis of scientific theories, ideas, etc.; aggregated information is transmitted using pedagogical software, e-presentations, etc.

The results of the distribution of three-subjects training, in the process of operation with information during a lecture are presented in Fig. 5.

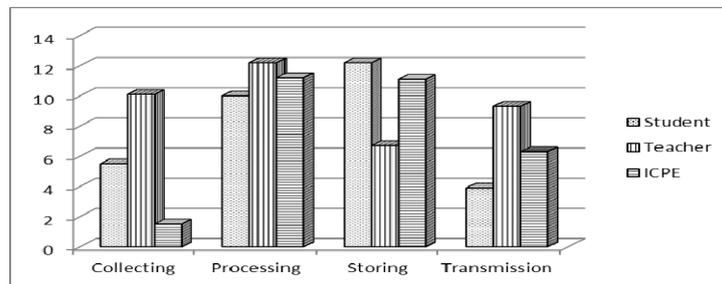


Fig. 5. The significance of the subjects of the educational process during operations with information on the lecture

Thus, the most important subject of the lecture organization according to students is a teacher (38%), a student (31.6%) and ICPE (29%) provide processing and preservation of educational information. In general, during the lecture principal place of work with information take processing operations (33,44%) and storing (30%), followed by transfer (19.5%) and collection (17.1%).

Let’s analyze the data from Table 4 according to the importance of the subjects of study during the preparation and conduction of practice. As it is known, practical lesson is a class that involves organizing teacher’s detailed study of individual theoretical positions discipline and development of skills in their practical application by individual performance to related tasks.

Analyzing the data in Table 4 concerning the collection of information (22.6%) during the practical sessions was revealed that students’ contribution is 7.7%, but teacher’s and ICPE respectively 9.2% and 5.7%. Comparing with a lecture, students’ activity increased by 2.2%, due to test theoretical knowledge of students, development of skills based on acquired knowledge and, as a result, a detailed collection of information for further processing.

In the process of collection (22.6%) and processing (26.8%) of the information during preparation and practice, according to students, a teacher and a student have the greatest significance, which is confirmed by received data: respectively (9.2% and 14.3%) , (7.7% and 8.7%). This is due to students interest in learning, deepening and refinement of knowledge, developing skills, primary accumulation of experience,

professional motivation and, consequently, activity in learning. Significance of ICPE is gradually increasing, as it is evidenced by statistics data, in storing and information transmission: 11.1% and 10.7%.

Visually, the results are presented in Fig. 6.

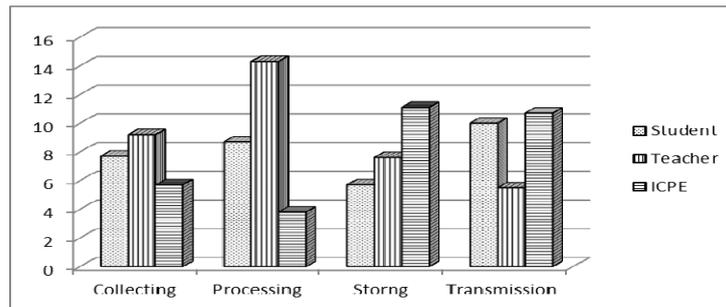


Fig. 6. The significance of the subjects of the educational process during operations with information during the practice

Thus, a teacher is the most important subject of practical training organization (36.6%), although a student (32.1%) and ICPE (31.3%) are equal subjects. Generally during practical classes the importance of operations with information is as follows: processing (26.8%), transmission (26.2%), storing (24.4%) and collecting (22.6%).

Let's analyze the importance of training subjects in the process of student's individual work organization. As you know, independent work of a student is a primary mean to master academic material at a time, free from mandatory training sessions.

Leading role in collection, processing, storing and transmission of material belongs to a student (33.9), according to the relevant data: 9.7% 10.4% 9.4% 4.4%. This is primarily due to the students' understanding of the importance of having theoretical knowledge, development of skills, and accumulation of their own professional experience and, as a result, operations with the information according to the educational goals. Practice has proved that the most active in independent work will be a student who is more motivated to master for his future profession. The result of questioning is the importance of teachers is on average 32.4%, ICPE - 33.7%, and an independent educational-cognitive students' work is task-teacher, under his leadership, but without his direct involvement but widespread use of information and communication teaching environment. The importance of business education in independent work is shown in Figure 7.

So, according to students' definition, important subjects of independent work are all three components of the didactic system - Student (33.9%) and Teacher(32.4%), and ICPE (33.7%). During the independent work with information, transaction processing occupies a principal place (31.5%), then collection (27.9%), followed by storing (27%) and afterwards transmission (13.6%).

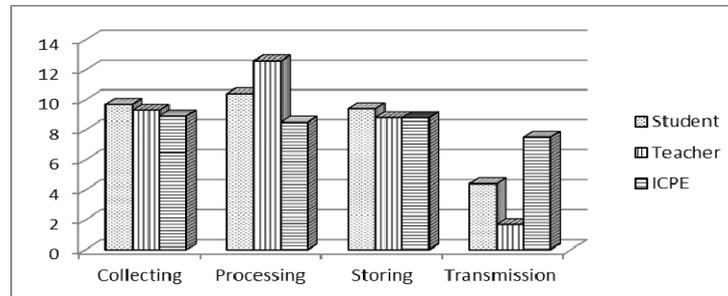


Fig. 7. The significance of the subjects of the educational process during information operations in class work

The analysis of the results of the survey showed that according to students' all three constituents are important and significant components of the didactic Student-Teacher-ICPE system. This is the statistics of indicators weight components: Student ($V_1 = 32,5\%$), Teacher ($V_2 = 35,8\%$), ICPE ($V_3 = 31,7\%$). It's important to underline that received students' survey data correlate well with similar data of experts' assessment of component's importance of the didactic system. The student is a significant subject of a teaching process at the University as learning outcomes largely depend on its intended acquisition of trade. Proof that serve high levels of significance to students in maintaining and processing information during lectures, collecting, processing and transmitting information during practice, collecting, processing, storing information during independent work. Major indicators of the importance of the teacher can be seen during collection, processing and transmission of information during the lecture, which is determined by specifics of this type of training sessions - teaching theoretical material. During practical sessions and independent students' work the teacher has the greatest indicators of the importance of information processing. ICPE's significance is high during operations with information and practical lessons in the process of information preserving, as for in-class, ICPE acts as an equal-right subject of the educational process. This is because ICPE provides access to informational resources at any convenient time, quickly and easily enables to find all necessary information, provides flexible and convenient information sharing between students. However, according to average data ICPE significance inferior teachers' importance, as a teacher manages the studying-cognitive students' activity, coordinates their independent improvement of knowledge, skills and abilities. It should be noted, that due to ICPE systematical involving in learning process the role of it as new subject will improve gradually because of improving learning outcomes.

Summary results of the survey are presented in Fig. 8.

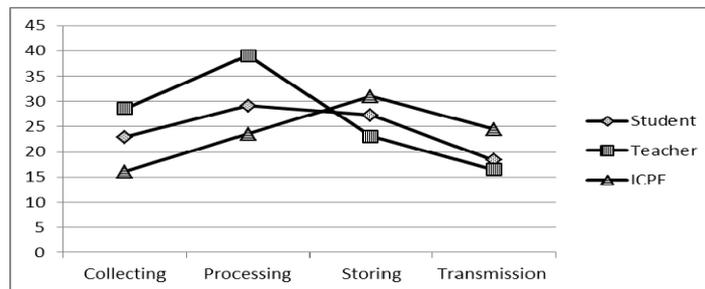


Fig. 8. The importance of the subjects of the educational process during operations with information

5 Conclusions

Thus, the analysis of the scientific literature, theoretical and experimental study on transforming learning into different didactic models showed that information and communication pedagogical environment is an important subject in the process of learning at the University; ICPE transforms traditional subject-subject model of study into three-subject one, directly affecting and slightly changing the role and function of other subjects of study, partly fingering their functions itself, particularly in transient conditions while performing operations with information on various forms of training.

References

1. Fokin, J. G.: Competencies of Education in University. Academy, Moscow (2002)
2. Kendall, H. L., Sugimoto, R. A.: The Didactic Theory of Wolfgang Ratke. California State University (1976)
3. Petukhova, L. E.: Theoretical Bases for Training Primary School Teachers in Information and Communication Teaching Environment. Scientific Monograph. Ayilant, Kherson (2007)
4. The Third Teacher: 79 Ways You Can Use Design to Transform Teaching & Learning by OWP/P Architects, VS Furniture, Bruce Mau Design, Abrams (2010)
5. Kravtsov, H. M.: Design and Implementation of a Quality Management System for Electronic Training Information Resources. In: Ermolayev, V. et al. (Eds.) Proc. 7-th Int. Conf. ICTERI 2011, Kherson, Ukraine, May 4-7, 2011, CEUR-WS.org/Vol-716, ISSN 1613-0073, pp. 88-98, CEUR-WS.org/Vol-716/ICTERI 2011-CEUR-WS-paper-6-p-88-98.pdf. (2011)
6. Rowe, G., Wright, G.: Expert Opinions in Forecasting: the Role of the Delphi Technique. In: J.S. Armstrong (Ed.) Principles of Forecasting – a Handbook for Researchers and Practitioners, pp. 125-144. Kluwer Academic Publishers, Boston, MA (2001)

Conception of Programs Factory for Representing and E-Learning Disciplines of Software Engineering

Ekaterina Lavrischeva¹, Artem Dzyubenko¹ and Andrey Aronov¹

¹ Taras Shevchenko Kiev National University, Kiev, Ukraine

lavryscheva@gmail.com, asmer@asmer.com.ua

Abstract. The paper presents a new idea of knowledge representation for students studying software engineering by developing artifacts and software components accumulated in libraries or repositories for further reuse. The idea is based on the concept of assembly line by V. Glushkov, further elaborated by Soviet and foreign specialists (A. Ershov, V. Lipaev, J. Greenfield, G. Lenz, Y. Bai, M. Fowler). The paper introduces the elements of program factory: reusable components, their interfaces, and assembly lines for designing and assembling complex software products from components. It is shown that modern operating environments provide prerequisites for such factories. The students' program factory, implemented at Taras Shevchenko Kiev National University, is then described. The technologies behind the factory, as well as its goals are studied.

Keywords. Artifact, reusable component, applied system, interface, assembling line

Key terms. Methodology, Technology, Process, Qualification

1 Introduction

The paper presents the elements of programs factories: reusable components (RC) and their interfaces, assembly lines for designing and assembling complex programs from RCs. It is shown that modern operating systems provide tools for creating specialized programs factories (MS.NET AppFabric, SOAFab, SCAFab, IBM VSphere, CORBA, etc.). Factories are built by different commercial structures for software product development, as well as for studying purposes (e.g., the programs factory implemented in Taras Shevchenko Kiev National University for the assembling artifacts from disciplines that are studied at the university). The purpose of such factories is to study computer science, software engineering, programming technology and software systems with electronic textbooks, develop applied projects and thus train highly competent professionals in software industry.

Over last decades a huge amount of various programs has been accumulated in the informational world that may be used as end products for complex programs devel-

opment. Therefore, a new approach has been formed in programming, namely reusability – reuse of ready-made software resources (reuses, assets, services, components, etc.), hereafter referred to as reusable components (RC). This term is used in informational world to represent new knowledge acquired over researches in certain fields of computer science. Being needed for somebody, it may be used in solving certain problems concerning similar artifacts as well as for development of new software systems (SS), applied systems (AS) or software product families (SPF).

All software artifacts and RCs may be stored in public warehouses (libraries, repositories) that may be searched by professionals to identify the necessary data for implementation in their own research activities. That is, reuse of ready-made resources becomes a capital-intensive activity in the field of software engineering and it is particularly important that universities possess so-called factories for scientific artifacts, programs and RCs needed by other students and professionals. With these factories, students may participate in industry development of scientific artifacts for mass use [1–3].

Based on such innovative ideas, Prof. E. Lavrisheva proposed fourth-year students at KNU to establish the first programs factory over the course of theoretical and practical labs on software engineering. This factory is focused on artifacts, software development, and repository maintenance. Students' programs factory operates on the web site (<http://programsfactory.univ.kiev.ua>) since December 2011. The web site has been visited by more than 5,000 people – students, scientists and teachers.

2 Establishing Programs Factory

History of Software Industry in USSR. An idea of industry for computers and supporting software has been formulated by Academician V. Glushkov at Cybernetics Institute of NAS of Ukraine in 1960s-1970s. Under his guidance, a family of small computers 'Mir' (1967–1975) has been developed together with a language of analytical and formula transformations for solving differentiation, integration and formula calculus problems. In addition, other computers (Dnepr, Dnepr-2, Kyiv-67, 70, macro-conveyor etc.) have been elaborated with auto code-typed programming languages (PL) to develop information processing programs and automated management systems (AMS) for various organizations and enterprises. For their development, the problems of software quality and increase of production of reusable components have been investigated with computers at state institutions [4–6].

In 1975 V. Glushkov first formulated the concept of assembling conveyor consisting from technological lines for software products (SP). The core of Glushkov's paradigm is to accelerate the transition from programming as an art to industrial methods of SP production in order to solve various economical, business, scientific problems with automated management systems.

Then (1978), software development as joint scientific-technical production and the projects requesting for automating SP creation have been decreed. The first pilot factory for software engineering was established for mass production of various AMSs (1978, Kalinin); but, because of lack of ready-made programs and immaturity of pro-

programming technology for industry production, the factory had lasted for two years and was closed [7]. Nevertheless the experiments aiming to elaborate programming automation tools were lasting until the collapse of the USSR.

V. Glushkov's idea concerning programs factories is now running at the several industrial factories explored by different authors theoretically and in practice [8–13]:

- Conveyor by K. Czarnecki and U. Eisenecker
- Software factories for assembling applications by J. Greenfield, K. Short et al.
- Continuous integration by Martin Fowler
- EPAM assembly line for building various types of software, improving software quality and reducing risk
- Automated assembling of the multi-language programs in heterogeneous environments by Y. Bai (VC++, VBasic, Matlab, Java, Visual Works, Smalltalk and others)
- Command development and assembling programs for software projects in MS Visual Studio Team Suite based on contracts
- G. Lenz's program factory utilizing UML in .NET
- Assembling, configuration and certification of global scientific-technical software on the European Grid factory
- Compositional (assembling) programming by E. Lavrischeva for developing software products from reuses, services, artifacts, and so on
- Experimental KNU factory

Careful analysis allowed singling out the key components of programs factories [13]:

- Prepared software resources (artifacts, programs, systems, reuses, assets, components, etc.)
- Interface as a mediator between two components, containing passport information for heterogeneous resources in a certain specification language (IDL, API, SIDL, WSDL, RAS, etc.)
- Operating environment with system facilities and tools supporting assembly lines with heterogeneous software resources
- Technological lines or product lines for mass production and assembling products.
- Methods of product development

The elements listed are the fundamentals of SP production industry at the factories that operate at major foreign software companies such as Microsoft, IBM, Intel, Apple, Oberon etc. At the KNU students' programs factory, these fundamentals are implemented as certain lines for programs and scientific artifacts, which are the best among student-developed programs factories.

The authors of the KNU programs factory consider it an integrated infrastructure for organizing production of mass usage SPs that are needed for customers and users from the fields of computer science, state government, commerce etc. The factory is equipped with technology lines (TL) or product lines [7, 12], as well as a collection of products, tools and services needed for automated processes execution over these

lines in modern operational environments (MS.NET, IBM, Sun Microsystems and so on).

Web Site of KNU Programs Factory. The main objectives for the web site are: improving students' skills in software development using the system for exchange of KNU students' certified software products and scientific artifacts; increasing SP quality and reliability; and learning to support the methods of SS industrial production.

The web site presents lifecycle models, SP building lines, the line for program production with the help of MS.NET platform, and examples of student programs. Obligatory requirements are maintained during certification to store software products in the repository of the factory (the library pool).

The main activities on the factory site are:

- Organizing program and artifact development
- Familiarizing students with tools and methods for program and SS development
- Representing students' software products in the repository
- Citing excerpts from articles and textbook materials concerning various disciplines

The factory is equipped with tools that allow broadening its functionality by specifying new software artifacts and storing them in the repository for further reuse. Each artifact is uniformly documented based on WSDL, IDL standard used in Grid global project.

3 Factory Lines and Components

Development of Technological Lines. Technological lines are created at the technological pre-production stage [7, 12] before SS production. They include activities for designing the TL scheme from processes and actions that determine the processing order of SS elements with appropriate technological modules (TM) or programming systems. The basic requirement of TL engineering imposed on production of programs and components is to assemble TLs from lifecycle processes meeting problem domain goals using standard tools, TMs, and the system of regulatory documents. The TL is then supplied with ready-made components, tools and instruments that generate and implement specific functions or elements, as well as the management plan for processes for changing states of the elements and providing quality evaluation [13-16].

The *RC model* for component-based development has the following specification:

$$RC = \{T, I, F, Imp, S\}, \quad (12)$$

where *T* is type, *I* is interface, *F* is functionality, *Im* is implementation, and *S* is interoperability service.

Basic operations over components are:

- Specifying components and their interfaces (pre- and post-conditions, which must be satisfied by the caller) in such languages as IDL, API, WSDL, etc.

- Maintenance of components, reuses and artifacts in the component repository for search, change and future integration into applied systems
- Integration of components into applications, domains, applied systems, software product families, etc.

All aspects of RC development and their usage in SP and software product families are goals for reusability disciplines and building material for these systems.

Applied system is a collection of software means (or functions of SS), including general tools (DBMS, protection systems, system services, etc.), constructed subsystems or components together with the tools for marshalling from one AS to another [14].

Product Lines at SEI. Product lines and product family (PF) is defined in ISO/IEC FDIS 24765:2009 (E) – Systems and Software Engineering Vocabulary: “*Product line* is a set of products or services that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way. Synonym: *product family*”.

SEI professionals propose the two models for representation of activities for SS development, namely engineering and process ones.

The engineering model assumes the three activities, namely RC development, PF development through RC configuration and management of the both above activities.

The activity for RC development presupposes PF scoping and production planning of SS collection accounting for the context of SS usage, production limitations and the chosen strategy. The PF development activity includes designing each specific SS implementation based on the set of developed RC, and building software systems according to the PF implementation plan. The management activity is based on balancing activities for RC development and PF maintenance tasks, and includes both organizational and technical management.

According to the process model, a set of processes is performed at the two levels, namely domain engineering, being also referred to as the development “for reuse”, and application (or SS) engineering – the development “with reuse” [15]. The last one is performed over assembly line using ready-made RCs to shorten time and increase SS availability. Therefore, configuring the product family from RCs according to the specific requirements and needs of a particular market segment is the final one in the cycle of production activities.

4 KNU Students’ Programs Factory

From the theoretical standpoint, program factories are based on the assembling conveyor that includes a collection of various more or less complex production lines for software artifacts, programs and RCs. Conveyor lines contain process execution using system tools or technological modules that automate process execution for obtaining interim or final results.

From the perspective of information technology, the factory provides the data processing toolset for the transition from individual programming of particular re-

sources to the industry of mass-usage SP. The factory increases SP development productivity during each lifecycle process due to use of RCs that possess the necessary functionality with due quality guaranteed by their developers. The assembling (composition, configuration) line may benefit through reduction of efforts because of use of readymade artifacts or RCs stored in the repository.

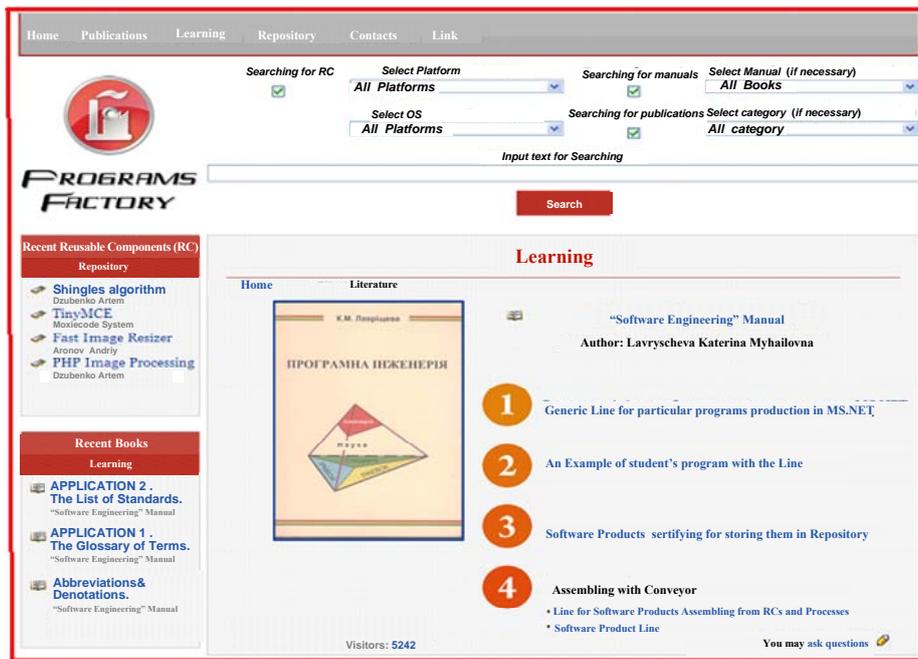


Fig. 1. Main page of KNU programs factory

Programs Factory Technological Lines. The TL development is as a rule matched with some lifecycle, e.g., implemented in the MS.NET environment with guides, frameworks, programming languages, common libraries templates, system tools that support new subject-oriented languages such as Domain Specific Language, etc. For complex SP development, an *assembly (compositional) line* is proposed, as a tool for composing RCs using their interfaces from various interim libraries within development environments, as well as from RC repositories (Fig. 1). The figure shows the main page of the web site of the factory: lines 1, 2, 3, 4 and a text-book for e-learning fundamental aspects of software engineering [2].

The assembling conveyor has four implemented technological lines [1, 2] that, since 2011, aid in artifacts and program development. The structure of these lines is designed according to the technology explored by the author (Prof. E. Lavrischeva) in 1987–1991 [7, 12-16]. Simple TLs in factory are as follows:

- E-learning C# in VS.NET environment (Fig. 2)
- Saving components into corresponding repositories and selecting them from the repository to meet market demands on specific SPs

- Assembling or configuring RCs into complex program structures (SP, FS)
- E-learning basic knowledge on software engineering with the dedicated e-textbook



Fig. 2. Chart for components design in VS.NET environment

Web Site Lines. The depicted TL for learning C# programming is designed according to the ISO/IEC 12207 standard of lifecycle processes while allowing for .NET specifics as to how to perform the following design tasks:

- Exploring demands on software products, singling out features and methods for automated generation
- Fixing requirements on implementations of SP functions for the domain
- Specifying software elements or artifacts, documenting their passport data and interfaces with a WSDL-like language
- Storing created software artifacts and programs in the repository

The line for repository maintenance includes the mechanisms for stored RCs being uniformly documented with their WSDL passports, as well as the tools for selecting ready-made RCs and artifacts from the repository based on their passport data, functions and relevant solution examples (see Fig. 3). This line is pertained to the processes for quality assessment of artifacts or programs created, verifying them from the perspective of reliability and quality.

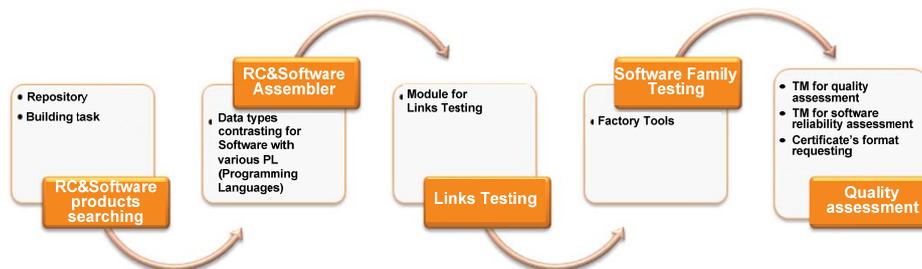


Fig. 3. Flowchart of the line for component search in repository

The line for AS development at the factory includes the tools for engineering of specific required components using various PLs, programs and artifacts, both just developed and selected from repository; the line also supports building multilingual RCs and marshalling non-relevant types of data exchanged by the components [11]. The constituents of this line include standard tools for building or configuring multilingual components, testing both the sample components and the links between RCs

being composed together, as well as the tools for reliability and quality assessment and certification of the product obtained.

The line for remote e-learning with the dedicated *Software Engineering* textbook [14, 15] is now actively used in studying topics of this discipline with KNU students. This line may also be used for independent studying in other high school institutions where software engineering or computer science courses are established.

Design of Programs Factory. At the students' factory, Visual Studio .NET licensed by KNU is used as the foundation for the programs factory; capabilities of MS.NET platform in providing tools for multilingual AS development and support using the components in C#, C++, Basic, etc., are utilized as well. Consequently, third-party software developers for the factory may not be limited to the choice of a single PL.

The factory web site is developed using PHP programming language, and, for its external representation, HTML5, CSS3 and JavaScript. The system core is independently engineered by the authors with relying on known web frameworks [2].

5 E-learning SE Disciplines

Teaching students the aspects of software industry at Ukrainian universities is at its initial stage. To solve several education problems, we have introduced a new approach to e-learning various aspects of SE, which assists in acquiring knowledge on software industry.

A new concept for breaking down the software engineering disciplines (Fig. 4), which is necessary in industrial factory production, was proposed [7, 14]. Basic goals of software engineering disciplines are as follows:

- Scientific discipline consists of the classic sciences (theory of algorithms, set theory, logic theory, proofs, and so on), lifecycle standards, theory of integration, theory of programming and the corresponding language tools for creating abstract models and architectures of the specified objects, etc.
- Engineering discipline is a set of technical means and methods for software development by using standard lifecycle models; software analysis methods; requirement, application and domain engineering with the help of product lines; software support, modification and adaptation to other platforms and environments
- Management discipline contains the generic management theory, adapted to team-based software development, including job schedules and their supervising, risk management, software versioning and support
- Economy discipline is a collection of the expert, qualitative and quantitative evaluation techniques of the interim artifacts and the final result of product lines, and the economic methods of calculating duration, size, efforts, and cost of software development.
- Product discipline consists of product lines, utilizing software resources (reusable components, services, aspects, agents, etc.), taken from libraries and repositories; it also contains assembling, configuring and assessing quality of software

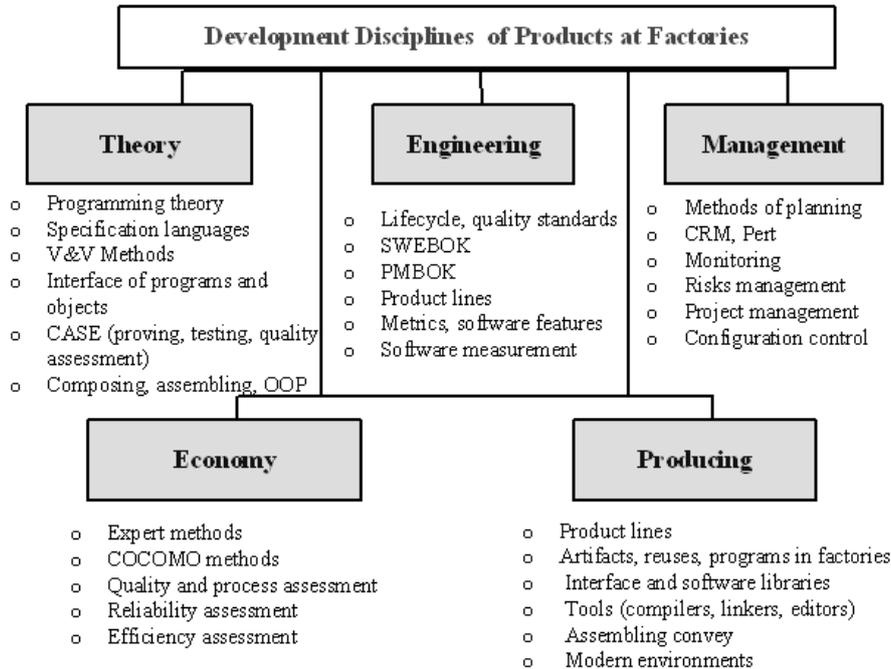


Fig. 4. Software engineering disciplines

In our opinion, all the SE disciplines considered above and their theoretical foundations must become the independent subjects to be taught to students specializing in the field of SE with the orientation toward the industrial production of SPs from readymade components (reuses, services, assets, and so on) [17]. The production cycle will be repeated in case of bringing changes in the product structure as it is done in technology of continuous integration by M. Fowler and in AppFabric in MS.NET (Fig. 5) [18].

Approach to Teaching SE. The SE educational course must include intertwined theoretical and practical fundamental positions and achievements in software development and integration [15, 17, 19]. The directions of the SE teaching course are as follows:

- Base concepts, principles and methods that constitute the basis of SE knowledge and technology of programming (e.g., the five SE disciplines, life cycle, project management, quality, configuration) and proved their productivity in practice
- Mathematics of systems analysis for subject domain with the use of elements of theory of algorithms, logic and semantics of programming for the formal design of key notions of the domain, reflection of their communications and relations in case of formal task of their models and SP architecture

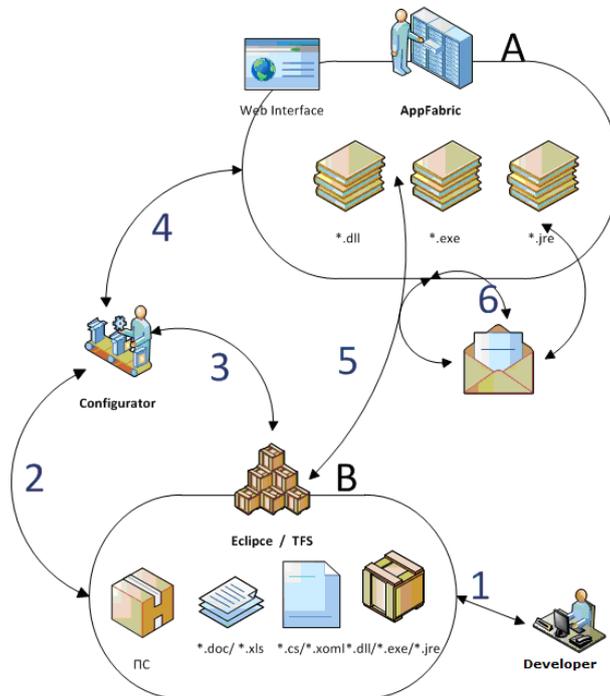


Fig. 5. Structure of Microsoft AppFabric

- General principles and methods of designing programs, software products and software product families using ready-made objects, components, services, aspects, etc.
- Modern applied tools for representation of software products, which are widely used by professionals in research and development (systems analysis, decompositions, architecture, design, OOP, ontology, etc.)
- Methods of measuring quality of software products
- Development environments (MS.NET, IBM, CORBA, Protégé, Eclipse, etc.)
- The directions of e-learning are summarized in the *Software Engineering* textbook. The students learn and apply them on practice with the use of system tools, artifacts or programs developed by other students. Certain students' achievements are certified by the teacher and can be added to the repository of the factory

In terms of teaching SE, the author's first textbook (2001), written in Ukrainian [18], is dedicated to the foundations of SE from Curricula-2001; the second textbook in Russian (2006) teaches the basics of Curricula-2004 [12]; the third textbook is developed for modern approach towards teaching SE [15], including topical outlines of some of the above-mentioned disciplines and fundamental aspects such as reliability and quality engineering. In the new textbook, basic elements and engineering tools are presented for development of various target SE objects and lifecycle processes, methods for design and management of collectives of executors, quality, terms, and

cost. The textbook on the web site describes new SE disciplines and fundamental aspects of SE. The structure of the textbooks corresponds to the typical Curricula-2004 program, and to the modern requirements on the subject imposed by the program of the Ministry of Education and Science of Ukraine (2007).

6 Conclusions

The result of the authors' work is an experimental programs factory web site, accessible on the Internet using address <http://www.programsfactory.univ.kiev.ua/>. This web site is proposed for e-learning product line development at the high school institutions on the specialties of informatics, computer sciences, information systems and technology.

The following concrete lines are established at the factory:

- Production of reusable components and artifacts
- Development of console applications, DLL component libraries, local Windows applications with C# in VS.NET
- Developing Java programs (a manual by I. Habibulin)
- Assembling programs from RCs in MS.NET environment
- E-learning software engineering with dedicated textbooks on the web site in Ukrainian (sestudy.edu-ua.net) and in Russian (intuit.ru)

The prospects of future factory evolution are its further adjunction with new resources in the field of software engineering being yet prepared by the students, namely:

- Description of the process of development of complex programs and SS using DSL language (Eclipse-DSL, Microsoft DSL Tools)
- Transformation of general data types into fundamental data types from the perspective of the standard ISO/IEC 11404-2007 generation tools
- Ontological representations of new disciplines for study (e.g., computational geometry, lifecycle domains, verification)
- New applied product lines for business developed with appropriate mechanisms;
- SEI product lines approach, and so on

References

1. Lavrischeva, E.: Concept of Scientific Software Industry and Approach to Calculation of Scientific Problems. *Problems in Programming*, 1, 3–17 (2011) (in Ukrainian)
2. Aronov, A., Dzyubenko, A.: Approach to Development of Students' Program Factory. *Problems in Programming*, 3, 42–49 (2011) (in Ukrainian)
3. Anisimov, A., Lavrischeva, E., Shevchenko, V.: On Scientific Software Industry. Technical report, Conf. Theoretical and Applied Aspects of Cybernetics. (2011) (in Ukrainian)
4. Glushkov, V., Stogniy, A., Molchanov, I.: *Small Algorithmic Digital Computer*. MIR, 11 (1971) (in Russian)

5. Glushkov, V.: Basic Research and Technology Programming. *Programming*, 2, 3–13. (1980) (in Russian)
6. Kapitonova, U., Letichevsky A.: Paradigms and Ideas of Academician Glushkov. *Naukova Dumka*, Kiev (2003) (in Russian)
7. Lavrisheva, E.: Formation and Development of the Modular-Component Software Engineering in Ukraine. V. Glushkov Institute of Cybernetics, Kiev (2008) (in Russian)
8. Czarnecki, K., Eisenecker, U.: *Generative Programming: Methods, Tools, and Applications*. Addison-Wesley, Boston, MA (2000)
9. Bai, Y.: *Applications Interface Programming Using Multiple Languages: A Windows' Programmer's Guide*. Prentice Hall Professional, Upper Saddle River (2003)
10. Greenfield, J., Short, K., Cook, S., Kent, S.: *Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools*. Wiley, Hoboken (2004)
11. Lavrisheva, E., Koval, G., Babenko, L., Slabospitska, O., Ignatenko, P.: *New Theoretical Foundations of Production Methods of Software Systems in Generative Programming Context*. Electronic Monograph, UK-2011, 67, VINITI RAN, Kiev, Moscow (2012) (in Ukrainian)
12. Lavrisheva, E., Grischenko, V.: *Assembly Programming. Basics of Software Industry*. 2nd ed. *Naukova Dumka*, Kiev (2009) (in Russian)
13. Andon, P., Lavrisheva, E.: Evolution of Programs Factories in Information World. *News of NANU*, 10, 15–41 (2010) (in Ukrainian)
14. Lavrisheva, E.: Classification of Software Engineering Disciplines. *Cybernetics and Systems Analysis*, 44(6), 791–796 (2008)
15. Lavrisheva, E.: *Software Engineering*. *Akademperiodika*, Kiev (2008) (in Ukrainian)
16. Lavrisheva, E.: Theory and Practice of Software Factories. *Software–Hardware Systems. Cybernetics and Systems Analysis*, 47(6) 961–972 (2011)
17. Lavrisheva, E., Ostrovski, A., Radetskyi, I.: Approach to E-Learning Fundamental Aspects of Software Engineering. In: Ermolayev, V. et al. (eds.) *Proc. 8th Int. Conf. ICTERI-2012*, CEUR-WS, vol. 848, pp.176–187, CEUR-WS, online (2012)
18. Kolesnyk, A., Slabospitskaya, O.: Tested Approach for Variability Management Enhancing in Software Product Lines. In: Ermolayev, V. et al. (eds.) *Proc. 8th Int. Conf. ICTERI-2012*, CEUR-WS, vol. 848, pp.155–162, CEUR-WS, online (2012)
19. Babenko, L., Lavrishcheva, E.: *Foundations of Software Engineering*. *Znannya*, Kiev (2001) (in Ukrainian)

Public Information Environment of a Modern University

Natalia Morze¹, Olena Kuzminska² and Galina Protsenko³

¹ Kyiv Boris Grinchenko University, Vorovskogo St. 18/2, Kyiv, Ukraine

n.morze@kmpu.edu.ua

² National University of Life and Environmental Sciences of Ukraine,
Heroev Oborony St. 16a, Kyiv, Ukraine

olena_k@bk.ru

³ Pecherska Gymnasium № 75, A. Ivanova St. 11, Kyiv, Ukraine

galinapro@gmail.com

Abstract. Processes of society globalization and technification require changes in training modern specialists and therefore involve changes of educational systems, including the creation of Information environments schools. The article is devoted to the topics of design, development and implementation of experience of Information environments in the educational process and scientific activities of universities. Developed by authors model of the environment has been implemented for constructing the information environment of the Kyiv Boris Grinchenko University and National University of Life and Environmental Sciences of Ukraine. The article describes approaches to the training of students and teaching staff of these universities for effective implementation and development of the resources of the Information environment. Conducted monitoring of resource usage of Information environment confirms the prospects of the authors' model and the methodology of its introduction.

Keywords. Information environment, experience, knowledge technology, learning platform, repository

Key terms. Environment, Academia, Development, Information Communication Technology

1 Conceptual Foundations of an Information Environment

Globalization of education leads to significant changes in the teaching systems: globalized learning goals, unified content and methods. New forms of technology and education focused on the integration of information and communication technologies (ICT) appear in the learning process. Especially significant changes occurred with the means of learning. From the concept of "learning tools" in the traditional model of education was made the transition to the educational environment in activity oriented teaching practice, then to the education space in the context of person-centered, individualized

approach, and finally to the Information environment (IE), which is realized in the process of development and ICT [1].

IE defined as a structured set of resources and technologies based on the consistent technological and educational standards which ensures free access of persons of the educational process to information resources, their effective communication and cooperation within such an environment for achieving educational goals that are known, understandable, achievable and concrete for them preliminarily.

IE of educational institution represents (has to represent) an adaptive model of global, national, information spaces and inherits their most characteristic functional properties, particularly in the communicative aspect of IE is a space of co-curricular activities based on ICT in the integration aspect provides the implementation of joint actions by establishing appropriate rules and the adoption of regulations that means that environment can emerge and develop only in accordance with the goals and objectives of the above-mentioned spaces, including the regulatory framework in the field of information policy at national and international levels, the state and prospects of development of information technology, the characteristics of learning process in educational establishment [2].

There is a list of computer technologies in annual reports of the International Media Consortium that will have (have) a significant impact on the organization of the educational process in the near future, namely: mobile technology and cloud computing (2009), open content, e-books , personal web (2010 - 2011 years), semantically compatible programs, Smart-objects, supplemented reality (2012 - 2014 years), educational games, sensor devices and interfaces, data visualization, training analyst (2015 - 2016 years.) [3].

In the world teaching practice Web 2.0 services are regarded as qualitatively new means of distribution and storage of teaching materials, effective tools of educational platforms [4, 5]. Wikis, blogs, social networks, websites and audio streaming, news channels allow users to collaborate - sharing information data store links and multimedia documents, create and edit content, solve practical problems, perform educational and research projects, etc.

That is why understanding the nature and objectives of the construction, using and developing of information protection, a clear understanding of its structure, components, systems development and selection of high quality resources, selection of effective service based on Web 2.0 technologies belongs to one of the main tasks of a modern University. With the current requirements of not only the operation but also the system of educational institutions, general management principles and principles of educational systems, as the leading principles of good design information and educational environment of the institution and of the overall architecture, should be allocated as follows:

1. *The principle of a systematic approach.* This means that build model should be based on systematic analysis of educational establishment. That means that structural elements, internal and external communications, which will consider the educational establishment as an open system, should be highlighted.

2. *The principle of modular structuring of information and data information.* The main purpose - to provide information and data needs in the most complete form, which allows to characterize the state of the system and provide adequate tools for the implementation of administrative functions and educational tasks.
3. *Principle of modification, addition and permanent renewal.* Implementation of this principle allows for expansion, upgrading and updating of the model with additional specific and understandable to persons indicators and measuring data. Thus, it can be changed or adjusted in accordance with the specific educational establishment, its traditions, mission and tasks.
4. *The principle of approximation,* which states that the system should be responsible for its complexity, structure, functions, etc. to those conditions in which it operates, and to those requirements that are set to it.
5. *The principle of giving the necessary and sufficient information* for the management of educational establishment.
6. *The principle of data sharing.* The same data can be used by several users. In addition, each user should receive this information in an easy to view it at any time and from any place.

2 Information Environment of a University – Concept Realization

Information environment of the institution at the present stage should include:

- Personal computing devices - a means of educational, researching and administrative activities of the institution
- Environment supporting collective and individual communication and cooperation
- Open educational resources - objects of educational activities and interactions
- Centralized and decentralized training platforms
- Means of information security and centralized filtration incompatible with the educational process content and more

The overall architecture (organizational structure and the associated operation of technological systems in education) is the basis of the process of creating educational technology systems adequate to the conditions of their use, in particular, an unlimited amount of resources that can be integrated into the educational process, a large number of users that can use the tools and technologies of technological systems, the number of students who may be involved in the joint solution of one educational task. Educational environment for such systems provides by international technological standards for interfaces, formats, communication protocols to provide mobility, interoperability, stability, efficiency and so on.

In this approach Internet is considered as a global platform of creation and dissemination of collective knowledge. Information environment is a mean and a place for creation, accumulation and harmonization of educational resources of efficient communication and cooperation, education and training of both students, teachers and administrators. The proposed model allows us to implement a set of technological principles

of open Information environment of the university such as adaptability, integrity, complexity, interoperability.

Construction of such Information environment provides a clear projection of its objectives, functional, access channels, organization of communication of students, teachers and researchers; system of continuous monitoring. The main features of the educational process in an open Information environment are:

- *Openness of environment* - students and teachers are actively participating in the development of educational resources and Information environment
- *Willingness of participants* - formation of need for building individual learning trajectories, positive motivation to cooperate and work in a team, willingness to disseminate the results of their own educational activities in the public access
- *Monitoring of objects and subjects of environment* - monitoring the quality of created resources, providing access to them and their efficiency of usage, observing the activities of the subjects of the educational process, organizing the feedback and assessment

Implementation of the proposed model in a particular educational institution involves the selection of platforms (Fig. 1) and resources:

- Scientific articles of educational-research and Masters members of National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- Abstracts of theses defended in National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- Conference proceedings of National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- Works of magisters of National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- Training materials to support the educational process of National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- e-Learning of National University of Life and Environmental Sciences of Ukraine <http://moodle.nauu.kiev.ua>
- Distance learning <http://agrowiki.nubip.edu.ua>
- Harmonized (supplemented by expert comments of National University of Life and Environmental Sciences of Ukraine and links to internal and external resources) standards <http://agrowiki.nubip.edu.ua>
- Regulations of National University of Life and Environmental Sciences of Ukraine <http://elibrary.nubip.edu.ua>
- Thematic practice-oriented information articles <http://agroua.net>

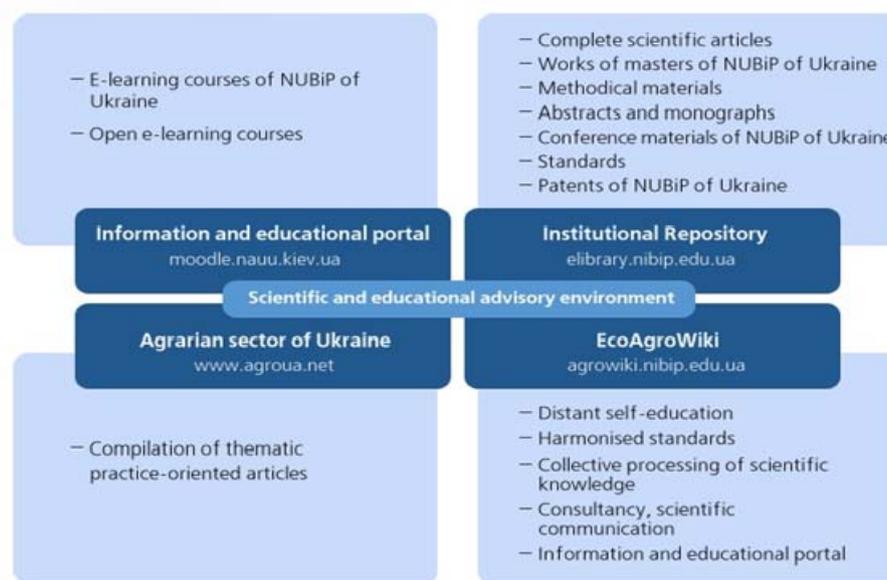


Fig. 1. Open Information Environment of National University of Life and Environmental Sciences of Ukraine

Effective usage of resources of the Information environment is largely dependent on the willingness of teaching staff to implement innovative pedagogical and information technologies and work with students in IE.

Question of training faculty can be solved on the basis of properly designed training system focused not so much on the study of specific technologies as on:

- Formation among the teachers methodical approach to the selection and usage in their professional activities IE resources to achieve educationally meaningful results in the context of ensuring the availability of educational materials, improving the quality and effectiveness of the educational process
- Developing skills of the educational process with the use of IE resources and managing innovative educational projects
- Logical design and creation of ICT-oriented learning tools

While building a training system for teaching staff to use IE resources in educational activities it is necessary to take into account the necessity:

- Modular structuring of content that reflects the technological and didactic possibilities of the usage of specific resources IE
- Malancing and harmonizing individual content modules training program for teachers

In the process of training teachers act as students (<http://lilia.moyblog.net/category/>) and this allows to simulate educational situations, identify problems and use of IE components to create training courses of new sample

(<http://moodle.kmpu.edu.ua/dn/course/view.php?id=144¬ifieditingon=1>). Topics of workshops:

- Institutional repository and its role in the creation of electronic educational and researching environment of the University
- Platform of e-learning Moodle
- Safe work in the Internet
- Creating of modern ontology on a base of wiki-portals
- Role of ICT in usage of formative assessment
- Usage of Web 2.0 for the students individual work
- Blogs and their usage patterns in the educational process
- Podcasts usage in educational process
- Role of ICT in the organization of cooperation and communication

Creation of Information environment will provide flexible formation of educational and methodical complexes according to the different models of learning, make teaching materials cheaper and more accessible, improve learning efficiency by providing sharing of experiences and a variety of educational materials between students and teachers. Organization of educational and research activities in the Informational environment determine what skills a student should possess, namely:

- Access to information data and resources – the ability to search, collect and store information data
- Management of information data resources – the ability to choose existing resources for categorizing and structuring information data
- Critical evaluation of information data and resources – the ability to make judgments about the data quality, importance, usefulness or effectiveness as well as the reliability, specific and address orientation
- Creation of information data - the ability to interpret and present data, generate data and knowledge
- Exchange of information data – the ability to transmit information data by means of information environment in a proper way

Acquiring of the mentioned skills and abilities is developed in the process of independent activity of students, such as the preparation and defense of Master's Thesis. The wiki-portal EcoAgroWiki is chosen as a technological platform of the informational support for Master's educational activity. Considering the functionality of wiki technology article authors managed to organize a community of masters, teachers, academic advisors practice (see Table 1). In order to simplify page layout on the EcoAgroWiki portal according to the standard IMS ePortfolio Information Model (http://www.imsglobal.org/ep/epv1p0/imsep_bestv1p0.html), it was designed templates of teachers and students portfolio. The results of network cooperation of its members is collections of useful links, digital electronic resources, target selection and description of the use of modern software tools, organization of project activities and collaboration, system of effective communication, consultation and expert evaluation.

Table 1. Subject of seminars series “Presenting of Masters’ scientific researches’ results using ICT”

Annotation	Resources
Topic 1. Electronic publishing	
Scientometrical bases (EBSCO). FAO- resources: AGORA.	http://web.ebscohost.com/ehost/search http://www.fao.org/index_en.htm
Topic 2. Institutional repository	
Institutional repository of NUBiP of Ukraine. OAI harvester of Ukraine. International repository.	http://elibrary.nubip.edu.ua http://oai.org.ua http://arxiv.org/
Topic 3. Bibliography	
Resources description variants. Personal bibliographic managers.	http://agrowiki.nubip.edu.ua/wiki/index.php/Zotero
Topic 4. Research results publication in the Internet	
Google documents, blogs, forums, conferences.	http://apctt.blogspot.com
Topic 5. Cooperation organization	
Google groups.Wiki-portal. University portal. LMS.	http://agrowiki.nubip.edu.ua http://it.nubip.edu.ua/ http://nubip.edu.ua

3 Experience of Resources of the Open Information Environment’s Usage by Students and Teachers

The real impact of the Open Informational environment into educational activities organization of the university was determined by on-line poll on wiki portal EcoAgroWiki and on-site discussions of the educational process. Master’s programme students of the NUBiP of Ukraine faculty of Computer Sciences and of the faculty of Ecology took part in the poll.

Before series of seminars authors examined the attitude of students towards implementation of the Information environment resources in teaching and research activities. The vast majority of students (189 of 200 respondents) believe that creation of Open Information environment significantly enhance the information support of education activities. However, 60% masters in computer sciences (hereinafter referred to as Group 1) and 20% of ecologists (hereinafter referred to as Group 2) consider effective the use of e-learning courses developed on the platform LMS Moodle, 50% of Group 1 and 30% of Group 2 use the institutional repository for viewing topics and presentations of Masters’ Thesis of previous years. The question with which search engines are students required information, including these with a scientific character, the overwhelming number of respondents named Google, and to save search results 70% of Group 1 and 50% of Group 2 use personal folders and file cards on personal computers. As of communication between students, most of them called social networks, and as of

communication with teachers (Academic Advisors) – personal correspondence via e-mail.

There are three the most important students' opinion, directions of the Information environment use. Students argued their choices from the position of information literacy [6]. 80% of Group 2 and 60% of Group 1 noted acquiring skills for analyzing the obtained data, sites, resources actively and productively; planning and managing their studying; establishing effective communication and cooperation; solving problems together, selecting the most effective resources and technologies to solve specific tasks.

Interviewed teachers (42 educational research worker of NUBiP of Ukraine) noted the stiffening of Masters' Thesis, especially in a part of analysis of research problem development, usage of modern resources, in particular materials of open scientific journals and bibliography description. In addition, Academic Advisors, who have joined the experiment, noted the increase of their own computer literacy through the usage of scientific communication means and cooperation in the process of joined work together with students. And creation of teachers' portfolio also make for promoting University activities, searching for partners in joint projects etc.

4 Conclusions

Experience of Information environment creation and usage of its resources in the Kyiv Borys Hrinchenko University and NUBiP of Ukraine suggests the following arguments in favor of the proposed model: individualization and personalization of the academic activity, quality, flexibility, ability to meet the educational requirements, timeliness, self and mutual control, cooperation. Open Information environment have to be built as a system of functionally and structurally interconnected information and technological elements, skillful usage of which allows a teacher in practice solve didactic tasks on the technological basis with a guaranteed quality in the age of education informatization.

Information environment creation at the level of educational institution leads to that educational materials and services will be available to every subject of the educational process. As a result, conditions for equal access to a quality education will be formed – an opportunity for everyone to learn at any place and at any time become a reality. Under these conditions, the Information environment is potentially unlimited as to the available quantitative and qualitative number of educational resources (can be used in the education process), number of users (can use its resources and technologies) and number of subjects of educational activities that can work together for solving educational tasks.

References

1. Manako, A. F.: Evolyutsiya ta Konvergentsiya Informatsiynih Tehnologiy Pidtrymky Osvity ta Navchannya. In: Proc. ITEA-2011, pp. 3–19, IRTC, Kyiv (2011) (in Ukrainian)
2. Bykov, V. Yu.: Avtomatyzovani Informatsiyni Systemy Yedynoho Informatsiynogo Prostory Osvity i Nauky. Zbirnyk Naukovyh Prats Umanskoho Derzhavnoho Pedahohichnoho Universytetu im. Pavla Tychyny, Ch. 2, 47–56 (2008) (in Ukrainian)

3. The Horizon Report: 2011 K-12 Edition, New Media Consortium, <http://www.nmc.org/pdf/2011-Horizon-Report-K12.pdf> (2011)
4. Bles, I.: Web 2.0 Learning Environment: Concept, Implementation, Evaluation. eLearning Papers, 15, 18, <http://www.elearningeuropa.info/en/article/Web-2.0-Learning-Environment%3A-Concept%2C-Implementation%2C-Evaluation> (2009)
5. Malinka, I.: Involving Students in Managing their Own Learning. eLearning Papers, 21, 13, <http://www.elearningeuropa.info/en/article/Involving-students-in-managing-their-own-learning> (2010)
6. Information Literacy at Otterbein College. Otterbein University, http://www.otterbein.edu/resources/library/information_literacy/index.htm

Designing Massive Open Online Courses

Vladimir Kukharenko¹

¹National Technical University “Kharkiv Polytechnic Institute”,
Frunze Street 21, 61002 Kharkiv Ukraine

kukharenkovn@gmail.com

Abstract. Connective massive open online courses (MOOC) for teachers from Ukraine and Russia were conducted in 2011-2013. They were: Strategy of Distance Learning in the Organization, Social Services in Distance Learning, Distance Learning from A to Z, Designing Online Courses. The accumulated experience allowed to develop recommendations for each ADDIE step of MOOC designing for the Russian-speaking audience.

Keywords. Connectivism, massive open online course, personal learning environment, ADDIE

Key terms. Competence, Didactics, TeachingMethodology, TeachingProcess, ICTEnvironment

1 Introduction

The term “massive open online course” (MOOC) was introduced by Dave Cormier in George Siemens’s distance course “Connectivism and Connective Knowledge” in 2008 and 2010 (<http://connect.downes.ca/>). This course was devoted to the problems of a new learning theory – connectivism, according to which learning is the process of creating a network (more than 2200 people studied). Units of such a network are external entities (people, organizations, libraries, websites, books, journals, databases, or any other sources of information). The act of learning implies the creation of the external network units. Over the last few years several dozens of open online courses have been conducted. Those courses are based on the new approach named “connectivism” and therefore abbreviated as cMOOC. cMOOC is characterized [1] by a structured network, the use of daily bulletin, a big amount of information material, the social approach to teaching. cMOOC enables people to cluster around the central core.

In cMOOC the teacher plays a lot of roles [2]: he is an amplifier, tutor, he directs and socially manages creation of meanings, he filters, models and is always present.

The student’s success in cMOOC is provided by his ability to navigate the network, the formed personal learning environment (PLE) and personal learning network

(PLN) as well as his personal goals. Personality development and personal learning play a central part in in cMOOC [1].

Experts believe that cMOOC [3] is suitable for effective independent learners, who have learned to select content. The reduced participation can be neither good nor bad. MOOC can be most effective as a form of continuous education and perfecting skills.

2 Analysis of cMOOC of National Technical University “Kharkiv Polytechnic Institute”

The courses are free of the traditional content. For each week is given an extended abstract and a set of links to various materials on the subject. A brief analysis of the topic is on the webinar, position of experts considered the on the guest webinar. Before the start of the course students are given instructions about the features of a connective course. Notes the desirability of selectively view recommended materials, preparation of remixes, posting material online and active participation in the discussion.

2.1 Distance Course "Strategy of e-learning Development in the Organization"

The open distance course "Strategy of e-learning development in the organization" lasted 6 weeks in February - April 2011 [4].

The main objective of this course was to show the possible uses of e-learning in the organization and to assist in the development of strategy of learning that takes into account the overall strategy of the organization; to learn the designing of the learning process in the open distance course, to assess readiness of the Russian-speaking audience to study in the new environment.

The target audience comprises teachers, post-graduate students, heads of educational departments from various organizations. The participants are expected to have skills of working in Internet, social networks and means of web communication (synchronous and asynchronous) for realizing communication, collaboration and exchange of information.

45 persons were registered for the course, pages of the course were visited by more than 100 people, 12 people passed the final survey. The questionnaire was completed by the same number of participants from the academic and corporate sectors with experience in educational work of more than 5 years (83%) and the experience of distance learning of more than 3 years (67%).

3 participants couldn't formulate their goal of taking this course, the rest of the participants (8 people) had the goal of getting acquainted with the features of open distance learning course and new social services.

The main activity of the participants could be traced by a mailing list (about 200 messages, half of them were sent during the first two weeks of classes, then the activity decreased); in Moodle only the forum of dating worked, all other invitations to discussions were not supported. Based on the materials of the course, two participants

created blogs. Generally, the group worked passively – its members read the proposed materials, did not disclose their sources, did not give their points of view on specific topics of the course.

There have been conducted six weekly, introductory and final webinars which were attended by about 10 participants each. All webinars were conducted in the environment of WIZIQ, one of them – in a virtual world “VAcademy”, where the participants got acquainted with virtual environment possibilities. Besides, 3 guest webinars also took place.

The experience of giving such course shows that the open course for CIS audience is a new and not always obvious concept, the great amount of instructional material and absence of clearly stated ideas cause great difficulties for participants. The limited set of social services, disrespect and misunderstanding of Twitter summons problems when tracing tutors’ and their colleagues’ work. Apart from it, the author thinks that the topic of the course was rather challenging for the learners. The fact that the personal instructional environment is not formed was the reason of problems arising during the learning process [4].

2.2 Distance Course “Social Service in Distance Teaching”

An open distance course “Social Services in Distance Course” [5] was held from May 23 to July 3, 2011 [6]. For this course Wiki, Mailing List, Twitter, DIIGO, learners’ blogs and aggregator netvibes.com were used.

This study examined the hypothesis that introductory webinars on forming personal learning environment (PLE) and the Workshop will increase the activity of students and material will be drafted for discussion by the full-time session. Besides, the introductory webinars were supposed to be held by a group of tutors.

Since the beginning of subscribing to the course active attendance of the course and surfing the pages started. By the moment classes began 43 people had been subscribed. At the beginning of the course we could observe the maximum of visits and browsing. Then the attendance was gradually decreasing. It should be noted that the number of visitors (Ukraine – 64%, Russia – 23%, the USA – 9%, Belorussia – 4%) was twice as big as the number of those registered.

30 people passed entrance questioning, 83% of them are university lecturers, the others are from the corporate sector; 74% of people have teaching experience of more than 5 years, 80% are experienced in making distance courses; 57% are experienced in tutor’s work; 37% use mobile phones to access the Internet.

It is essential that personal learning environment of the course’s participants is poorly formed, only 10-20% of participants use most services which is not helpful in communication.

Most of the participants stated the aim of the training as a necessity to master the new social services and outline the new approaches to distance training on their base.

On completing the course 10 participants took part in the questioning. On average the learners spent 6-8 hours a week.

2.3 Distance Course “Distance Course from A to Z”

The open distance course “Distance Course from A to Z” includes two parts. The first part which was held from December 5, 2011 to January 22, 2012 was devoted to the tendencies of creating the system of distance training at the current stage of Internet development. In the second part - the distance course design methods and the distance learning process.

The main objective of the course is to analyze the level of distance training development in Ukraine on the base of webinars held in May-October 2011 [7], consider the tendencies of distance training development abroad and outline the requirements to the modern system of distance training.

The course is based on the blogs and articles published 2-3 months before the course started. Such references show modern tendencies in the distance learning system mainly abroad. To shape the references Twitter was used.

Before the course started the following webinars were held: “Social Services in Teaching Process”, “Twitter”, “Personal Learning Environment”. Their task was to help the learners acquire the skills of using social services during the course.

The total number of those registered in the course is 31. According to Google Analytics the course was attended by 430 people from 29 countries of the world, 117 cities. Among them: from Ukraine – 76%, Russia – 13%, Belorussia – 5%. On average there were 30 people a day. Surfing the weekly pages ranges from 550 to 200. During the course 12 people wrote 68 blogs, 85 messages were left in the course. More than 80% of participants have teaching experience of more than 5 years, 68% of them – more than 10 years. The experience in distance course usage is from 1 to 10 years.

22 participants answered the question of the final questionnaire. They are experienced teachers with 10-year experience of pedagogical work and 5-year experience of distance training, half of them worked near 3 hours in the course (3 persons worked more than 8 hours). During the course most of the participants stated their aims, which can be generally outlined as follows: to obtain and systematize the ideas of modern distance training and other universities’ experience.

Answering the question about the novelty of the distance course all participants mentioned such services as Twitter, DIIGO, Creative Commons licences, personal learning environment, open learning resources and open distance courses, new approaches in distance education.

The participants of the course liked the system approach to the problem, the format of the course, unobtrusive and not strict management, great number of various information resources, possibility to work with information and new services in a suitable time, openness of communication, active information and experience sharing, the potential efficiency of the course, as under certain conditions the mechanism of self-reproduction of the course can be launched, that is, it will start working without visible participation of the organizers.

The learners noted the challenges of the open distance course, such as mastering new tools, personal aim of the study (dynamic and not always concrete), work with great volume of unstructured information in foreign languages.

The participants' work and their impressions of the training process of Mass Open Distance Course were considered at the Xth International Seminar "Modern Pedagogical Technologies in Education" which took place on January 31 – February 2, 2012 [8]. One of the suggestions was to organize the groups by interests.

It should be noted that the attendance of courses did not stop after they were officially closed in April 2012.

On the whole during the year the course was attended by 2492 users, most of them live in Ukraine (Kharkiv, Kyiv, Odessa) – 60%, Russia (Yaroslavl, Moscow, Yekaterinburg) – 20%, and Kazakhstan (3%).

2.4 Distance Course "E-learning Design"

In 2012-2013 academic year the Research Laboratory of Distance Learning made an attempt to hold a combined MOOC course "E-learning Design", which consists of a course for beginners (xMOOC constructivism approach) and that for heads of distance learning centers of an organization (cMOOC connectivism approach). The free program of the course includes three modules: "Basics of Distant Learning" (6-week long), "Technology of the Development of Distance Learning Course" (12-week long) and "Practicum for Tutors" (6-week long).

The objective of the course is to improve and standardize the level of teacher training in distant learning for colleges and universities in Ukraine.

Experienced professionals (managers) of distance learning take the course, built on the connectivism principle, in order to organize education for distance learning course developers in their organizations. The educational process of the course encourages the exchange of experiences among the students, improving the quality of the course. Students plan the training for distant-learning course developers either independently or making use of the proposed distance course for beginners. The overall goal of this course is to improve the efficiency of training for distance learning course developers. To participate in this course, you should be able to use social services (twitter, RSS, netvibes, paper.li, scoop.it and others) and log in the open course [9].

The second course, built on the principles of constructivism, is where beginners learn to create their own distance courses. The tutor provides general management of the educational process, the leaders of distance learning centers may act as local tutors for their teachers. If desired, these students can take part in the first distance learning courses as well.

As many as 90 individuals from the academic (91%) and corporate (9%) sectors, with distance learning (45%) and tutor (46%) experiences logged in the course. Most of the students (64%) have professional experience of over 5 years. Most students (43%) spent over 5 hours each week working on the course, 3-5 hours per week (31%), with the rest spending less than three hours weekly (25%).

The intention of the students involved were focused mainly on observation of course events (80%), participation in discussions (75%), establishing new contacts (72%), creating a distance course (68%).

After the first distance-learning course "Basics of Distance Learning" the open connective part of the Wiki course was closed because of student low activeness while the course participants were active in making their assignments.

The second part of the course "The Development of Distance Learning Course Technologies" envisaged the involvement of groups of distance courses developers from various universities together with their instructors. However, only the teaching staff of Kharkiv National Pharmaceutical University accepted the invitation. During the educational process, they were very active in various forums, helped each other and created distance learning courses.

Among 61% of the logged-in students involved in the training, 40% worked actively, yet only 11% (9 people) complied with the course program.

Apart from creating distance courses, the students filled out workbooks (5 tasks, performed by 29...12 participants), questionnaires (6 tasks, done by 34...12 participants), did two tests (32 and 27 participants), and discussed various issues in 10 forums. More than 5 hours per week were spent to work on the course by 43% of students, from 3 to 5 hours a week – by 31% and 25% - less than 3 hours a week.

3 cMOOC Design

The courses held allowed recommendations on the ways to design cMOOC for Russian-speaking audiences. It should be noted that currently the MOOC design is being focused on technical aspects, i.e. the choice of content media placement, communication, aggregators and other services. Stephen Downes [10] has given some recommendations for cMOOC design.

MOOC design is supposed to use the ADDIE (Analyzing, Designing, Developing, Implementing, Evaluating). This approach is an adaptation of the design methods of technical facilities for the pedagogy.

3.1 Analysis

cMOOC features uncertain audience and a variety of learners' purposes. Therefore themes of a distance course are determined by the author of the course, or, which is preferable, by the author and his/her team.

The themes must be relevant, contemporary, and a lot of unstructured information should be generated in the selected area. A theme is the best choice if it is the object of author's research. In this case, the author can formulate his/her research aims, this leading to the program of the course.

The main problem at this stage is a small audience and its inactiveness, poor PLE. Most often, students are targeted at getting acquainted with some information and establishing new contacts. A course participant must have essential implicit knowledge on the theme of the course, technical skills to work with social services and time management tools.

3.2 Design

After completing the course program, the duration of the course should be determined. The course duration is desirable not to exceed the time period of 6-8 weeks as students find it difficult to concentrate for longer periods. The next step is to draw up draft descriptions for sections of the course.

A most critical step is to select informative materials. To do that, the author of the course should possess the content curator skills. In this case, a vast number of links on Twitter to various information resources treated as blogs, e-magazines like scoop.it and much more are made available by the time the course is ready, which is among the content curator's function.

The most recent links to informative materials should be selected; and it is desirable to choose the number of links on the topic that are above Dunbar's number in order to organize selective information processing for students. Dunbar's number is a cognitive limit to the number of people with whom one can maintain steady social relationships (100 - 230, 150 selected) [11]. With a small number of links, students psychologically tend to seek reading all materials.

After selecting the informative materials it is advantageous to write a brief abstract for each link, e.g. to use the tools cruxlight.com, and sort them out according to areas so as to make students' work easier.

3.3 Development

Now comes the time for a summary for all practical hours, with problem-setting and specifying various tasks for students. A list of recommended student activities is desirable, e.g. to make about 10 retweets, write 2-3 blogs with a review of some sources etc. The course should contain a variety of recommendations and references for those who have little experience of using social services.

It is also necessary to develop entry and graduation questionnaires. Weekly questionnaires are recommended to include questions:

- Which materials were of interest? Why (not)?
- What was new and interesting in the webinar?
- Which can be considered an alternative blog?

Students can choose between filling out the questionnaires and writing a blog (re-mix, reflection).

The next step is to select guest lecturers for each week. They can be author's colleagues or involved students who are leading experts in some aspects of the course.

3.4 Implementation

MOOC should be conducted by a team of tutors. Their roles may be different. For example, they can share functions determined by the main author of the course. The tutors can be independent, each of them preparing their own materials for the course and giving his/her opinions during webinars without prior consultations. The most

important thing is that teams of tutors work very actively thus setting the work pace for the audience.

Before the MOOC starts, it is advisable to hold 1-2 webinars telling about the tools employed, especially twitter, evernote, mailing lists, blogs, and use of translators. This could be helpful for students to get prepared for the educational process.

The process of designing a distance course is an iterative process of creating a fundamentally new service. It requires studying other authors' experiences, new social services, continuous work with new information related to course themes.

4 Resume

The experience shows that it is very difficult to introduce cMOOCs in education and distance learning in the CIS. This could be due to the choice of themes for courses, low activeness of the pedagogical community, a small number of social services used. For example, Twitter is not very popular in the educational community. At the same time, four courses held led to formation of a community of as many as 30...40 members who actively participate in all cMOOCs organized not only in the CIS.

Low activeness of cMOOC participants does not allow implementing the principles of connectivism. Therefore, open courses are first needed on the use of social services in education, followed by training content curators. Furthermore, specific features of CIS audience must be considered when designing cMOOCs.

References

1. Downes, S.: Education as Platform: The MOOC Experience and What we can Do to Make it Better. March 12, <http://halfanhour.blogspot.com/2012/03/education-as-platform-mooc-experience.html?sref=tw> (2012)
2. Bosman, J.: Teacher Roles and MOOC, <http://moocblogcalendar.wordpress.com/2012/03/19/change11-teacher-roles-and-mooc/> (2012)
3. Stevenson, D.: MOOC- The Recent Discussions about MOOCs. <http://learning-aworkinprogress.blogspot.com/2012/03/change11-mooc-recent-discussions-about.html> (2012)
4. Kukharenko, V.N.: Innovation in e-Learning: Massive Open Online Course. High Education in Russia, 10, 93–98 (2011) (in Russian)
5. Distance Course: Social Services in Distance Learning. <http://el-ukraine.wikispaces.com/> (in Russian)
6. Kukharenko, V.M.: Learning Process in Massive Open Online Course. Management Theory and Practice in Social Systems, 1, 40–50 (2012) (in Ukrainian)
7. Webinar Records of Chief of Center of Distance Learning Ukraine Universities. <http://dl.khadi.kharkov.ua/mod/resource/view.php?id=8404> (in Ukrainian)
8. X Workshop Materials, <http://dl.kharkiv.edu/mod/resource/view.php?id=11229> (in Ukrainian)
9. Distance Course: Design e-Learning. <http://de-l.wikispaces.com/> (in Russian)
10. Downes, S.: Creating the Connectivist Course. <http://moocblogcalendar.wordpress.com/2012/01/03/creating-the-connectivist-course/> (2012)
11. Wikipedia: http://en.wikipedia.org/wiki/Dunbar's_number (2012)

The Role of Informatization in the Change of Higher School Tasks: the Impact on the Professional Teacher Competences

Dmitry Bodnenko¹

¹BGKU, Borys Grinchenko Kyiv University, 18/2 Vorovskogo st. 04053 Kyiv, Ukraine

bodnenko@kmpu.edu.ua

Abstract. Last decade is characterized by the tendencies of the modern higher school development. This is based on the informatization of education. In this work, basing on didactics of higher school pedagogy a system of psychological and pedagogical characteristics for higher school teachers was created. Psychological, educational requirements for professional competence of university teachers are based on components of pedagogical skills of teachers of higher education institutions and absorb almost all of its functions, duties and skills, but increased usage of ICT determines the specification of the classification system of psychological and educational requirements for information and communication competence of the university teacher.

Keywords. TeachingMethodology, competence, distance education, technologies of distance education, tutor, listener, course of distance education, informative resource, network services

Key terms. Didactics, CompetenceFormationProcess, InformationCommunicationTechnology, TeachingMethodology

1 Introduction

In the conditions of the pedagogical paradigm updating, emergence and distribution of network technologies, and consequently, enrichment of personality aspects of modern teacher preparation in higher school, a large value is acquired by interpretation of the teacher's professional competence concept.

The problem explored by us in this article is concerned with how the informatization of education influences the requirements for professional competence of teaching at universities.

During the recent years, in Ukraine ICT development has been defined as one of the priorities. In 2007 the Parliament passed the Law "About the Basic Principles of information Society's development in Ukraine in 2007-2015" (№ 537-V of 9.01.2007), pursuant to the Action Plan has also been adopted. Both documents were designed to promote the development of information society and the introduction of

information technology as apriority direction of the state policy. The need for further development and implementation of ICT is confirmed, also, by a number of national documents, such as the Draft of National Education Strategy in Ukraine for 2012-2021 years CMU (Meeting of 11.09.2012), the Laws of Ukraine "About the conception of the national programme of informatization" (№ 75/98-VR of 04.02.1998) with amendments introduced according to the laws N 3421-IV (3421-15) of 09.02.2006, VVR, 2006, N 22, article 199, N 3610-VI (3610-17) of 07.07.2011).

Unfortunately, these intentions, mostly in the present time, are remaining on an embryonic stage, that is reflected in the low Ukrainian ratings of competitiveness and network readiness.

Implementation of informatization at the universities hasn't been explored fully yet. This topic opens up many opportunities for further studying and investigation: implementation of educational programs, websites, the use of network services, an invention of new forms, ways, exploring of university teachers potential at the new open education environments. Writing this article we set the following goals: exploring of the university teachers' professional competence; the definition and justification of psychological and educational requirements for information and communication competency of University teacher.

2 The Main Part of Our Research

In recent years, the term "tutor" became well-known, it gained a considerable popularity in higher education, giving ground in frequency of use only to the term "teacher". These notions are almost synonymous, which could cause some dubieties about the necessity of introducing the new foreign term. But, in fact, the token tutor expands the notion teacher, especially in the context of the gradual integration of Ukraine into the European educational space.

S. Goncharenko [4] notes that the term tutor (English tutor from Latin *tueor*, to observe, to care) – is a teacher-mentor in British "public schools", high forms of grammar schools and teachers colleges [4].

In modern educational paradigm, regardless of the learning forms, the principle about the student's importance was established, that fact illustrates a student as a dominant. According to this fact, teachers act as assistants, as friend, as mentor, who supports students in getting education. With the rapid growth of informational and communicational component of the educational process, particularly in the context of the implementation of distance studying, it is difficult to imagine a teacher, who just transmits the information to the listener, even if it is a video lecture. Teachers need to be a coordinator, facilitator, who "synthesizes and accompanying student's resources" [6], which has much bigger freedom of the choice (educational content, time, place, methods of education), compared with the traditional student.

We support the aspects of the network training's specific which is - pointed out in V.M. Kukhareenko's and V.Y. Bykov's studios [5], which claim that the teacher-tutor should focus on his practice, doing his classes for the listener of the e-learning course.

However, working in the field of distance studying, the teacher communicates with the diverse contingent.

Each person is a personality, according to some needs, abilities and opportunities. That is why the teacher's task is to choose the best way of the studying process. They have to coordinate their activities (growth, educational content, methods, tools) with options of the audience potential or take responsibility for themselves, creating a new pace of studying. Also they should to select, a group of listeners who exactly overtake the course (it is not the fact that all students will be able to take this rate).

As the research is based on the condition, that the teachers of the higher educational institutions are studied for the implementation of the e-studying course, so we agree with S.S. Vitvytskaya's supposition [3] that teachers have to learn to perform the following functions: organizational (the head, the leader in the maze of the knowledge and skills); informational (the carrier of the last information); transformational (transformation the socially meaningful content of the knowledge to the act of the personal knowledge); orientational and regulative (the teacher's structure of the knowledge determines the structure of the student knowledge); catalytic (transformation the object of the education into the subject).

The teacher, who is not at the top level mastering pedagogy of higher education, it is difficult, in our view, to prepare for the implementation of distance learning in higher education.

According to A.I. Kuzmyskyi [7], the higher school teachers have some competence (the high professional competence, pedagogical competence, social and economic competences, communicative competence), have a high level of general culture and, also we can highlight especially important, in our mind, for the introduction of ICT competencies the functional responsibilities the universities' teachers, in particular. All these ingredients along with the pedagogical skills (except some components of the educational technology), outlined in the A.I. Kuzminskoho's works [7], form the basis of psycho-pedagogical portrait of the teacher, and transforming the system MDs we will get in the future - a distance studying tutor or teacher, who has information and communication competences, who is ready to work upon condition of a new paradigm of education. In particular the pedagogical skills include: moral and spiritual qualities, professional knowledge, social and pedagogical qualities, psychological and pedagogical skills, pedagogical technique.

Russian authors, as G. Adrionova [1], M.V. Vislobokova [11], V.P. Verzhbytskyy [2] say that the traditional teacher and teacher of the e-studying - are mutually different personalities with the different characteristics. We disagree with this opinion, because with the help of the teaching skills of higher school teachers (as one of the key characteristics of the higher school teacher) it was possible to form the teacher - oriented on the using of new teaching technologies, including ICT. The fact that many characteristics are really opposite, because they depend on the tasks set for the teacher and the student.

The most part of the scholars and teachers asserts [8, 9]: teacher - is a basis of the educational process, the most important component, that organizes the e-studying process and ensures its quality. V.M. Kukharenko [5] emphasizes the role of the tutor

in the e-studying system: "Any course requires the tutor, but a good course - requires skillful tutor".

Distance studying technologies that are introduced at the time of the rapid development of interactive technologies, absorb current dominant of educational paradigm: the activities of the teacher-tutor designed to organizing, promoting and supporting of the students' independent learning activities at the distance learning courses, which has the development of creativity as a basis, developing of searching abilities, analyzing and organizing information and rendition on the basis of the the right decisions' findings. In our opinion, to develop the creative abilities of the listeners can only the teacher, who also is a creative person.

In the context of the above statement is A.M. Eagle's saying [10] that the tutor – is a teacher of the high level, who is able to interact with the audience, producing new relaxed and fun lessons for all the. The researcher says that the tutor doesn't have to teach the audience, he has to maintain him as long as the student takes sufficient independence and competence.

The proof of this distribution is the work of the authors from the Problem Laboratory NAM NTU "KPI" [5], where the experience of the foreign researchers [12,13] is adapted to the Ukrainian Distance Learning System and where is outlined some tutor's responsibilities, according to two stages (development of the course and organization of the educational process). We have to note that in this case the tutor can be a user of already created e-studying course.

So systematizing the lined material, taking it as a basis, based on the generally accepted didactic principles, which are clearly defined in the A.I. Kuzminskiy's work [7], form the system of psychological and educational requirements for informational and communicational competences of the university's teacher.

Informational and communicational skills include the following components:

- To have at least one information-educational environment
- To know the range of services provided by the environment and technology of the handling these services;
- To know the basic principles of the telecommunication systems
- To know the specifics of the webinar, audio, video- teleconferencing, chats and forums;
- To know the rules of conducting (etiquette) during the interactive dialogue
- To know the specifics of working within formational resources (databases, information services)
- To be able to use the communication capabilities of computer' networks to organize fruitful communication between the participants of the educational process
- To make the organization and conducting of the telecommunication project
- To own and use network services in a professional activity

Didactic skills:

- To create and shape the course material for students of e-learning courses with the optimal (understandable, accessible, scientific) laying out the information to ensure personal, effective and independent from the listener's time and his working place
- To implement psycho-pedagogical monitoring (previous, current, interim, final)

- To manage the independent educational and cognitive activity of the students, to develop intellectual capabilities and to form the motives of the education
- To teach students the efficient and effective methods of independent activity in the educational process
- Constructive skills:
 - To integrate and combine full-time, part-time, external and distance learning
 - To create e-learning and /or correct an existing course, according to the educational process requirements
 - To adopt the effective types and forms of participants' activity of the e-studying/network educational process
 - To make the selection of the methods and means of education
 - To have the skills of the informational navigation
 - To plan the perspective stages for the management of the students group (small group)
 - To organize using the distance learning technologies of the individually-oriented approach to the audience
- Organizational skills:
 - To balance the demands of discipline with the students' needs
 - To demonstrate to the listeners their personal potentialities concerning to the provided educational information
 - To carry out a systematic discussion about the students needs for continuous improvement of the distance learning process
 - To provide the necessary support and assistance to the ENK students;
 - To organize and conduct network role games
 - To organize and manage the students' activity, in a small group, to create the optimal conditions for the development of their independence and competence and provide pedagogically effective activity of the listeners
 - To organize the participants' meetings of thee-studying process
- Cognitive skills:
 - To study physical, psychological and social components of special features of listeners' individual development, their needs, social self-determination etc.
 - To analyze individual styles cognitive-educational activity of listeners
 - To involve modern pedagogical (tutor) experience (learning in cooperation, small groups method, project method, different-level education, forming evaluation, research, explore methods etc) and creatively apply it in own tutor practice
 - To master new scientific information in the subject field, methods of teaching and use rationally in scientific-pedagogical work
 - To search for facts that stimulate activation of the cognitive activity of students in informational and educational environment and apply them
 - To generate new ideas and perspectives of tutor and student activities and apply modern technologies, forms and methods of distance learning
- Communicative skills:
 - To determine the feasibility of the relationship between subjects of educational process

- To coordinate interpersonal relationship between the students in the group or between students in small groups
- To prevent conflict situations that may arise in the process of e-studying, resolve them
- To apply collective activity, cooperate, determine common strategy of activity and prove its relevance, be able to admit own mistakes

Make simple, easy and tolerant communication with any age, social and ethnic categories of students.

Perceptive skills:

- To understand (by the look of students) incentives of activations of students' activity in information-educational environment and be able to apply such knowledge
- To be concerned with the inner world of the audience, understand their mental state. Observe special features of the independent activity of the student in an Information-educational environment during e-studying process.
- To improve directly the technology, information saturation, activity, depending on the needs of the group (the audience)

Suggestive skills:

- To master a method of forming a systematic and critical thinking
- To form reflection in students as a mean of evaluating their activities with the purpose of further improvement
- To affect (emotional and volitional aspect) students with forms, methods and means of e-studying to create in them a certain mental state, prompting them to definite actions

Applied skills:

- To possess additional hardware, software, psychological educational equipment
- To create Web Pages, publications, websites, blogs, wiki, etc.
- To have skills to program in specialized environments

Skills in psycho-technical sphere: knowingly and properly use acquisitions from psychology in field of applying network services.

3 Conclusion and Future Work

Psychological, educational requirements for professional competence of university teachers are based on components of pedagogical skills of teachers of higher education institutions and absorb almost all of its functions, duties and skills, but increased usage of ICT determines the specification of the classification system of psychological and educational requirements for information and communication competence of the university teacher.

According to the results, which we obtained during the research, we can outline some basic theoretical and empirical achievements of the author in the context of the study objectives:

Professional competence of teachers of higher education on condition of higher education tasks' changes and transition to the society of knowledge were outlined.

System of psycho-pedagogical requirements for information and communication competence of university lecturer was formed and explored. The system consists of a

list of skills: informative-communicative, didactic, constitutive, managerial, cognitive, communicative, perceptual, suggestive, applied and the skills of a psychotechnique sphere.

Prospects for further scientific research are seen in detailed usage of professional competence of university teachers in practical activities, including the usage of network services in teaching students of Humanitarian specialties.

References

1. Adrianova, G.: *Typologiya Subjectov Distantcionnogo Obucheniya*. In: Proc. All-Russian Remote Teachers Council (2000) <http://www.eidos.ru/books/read-room/andrianova1.htm>. (in Russian)
2. Verzhbitsky, V.P.: *Distance Education in Russia*. <http://www.tcde.ru/de/st001.html> (in Russian)
3. Vitvitcka, S.S.: *Osnovy Pedagogiky Vyschoi Shkoly*. Textbook. Second Edition. Center of the Education, Kyiv (2006) (in Ukrainian)
4. Goncharenko, S. V.: *Ukrainsky Pedagogichny Slovnyk*. Second Edition. Lybid, Kyiv (1997) (in Ukrainian)
5. Bykov, V. Y., Khuharenko, V. N. (eds.): *Distance Education Process*. Textbook. Millennium, Kyiv (2005) (in Ukrainian)
6. Koycheva, T. I.: *Preparation of Future Humanities Teachers as Tutors of Distance Learning Systems*. Manuscript. Konstantin Ushinsky South Ukrainian National Pedagogical University, Odessa (2004) (in Ukrainian)
7. Kuzminsky, A. I.: *Pedagogika Vyshoi Shkoly*. Textbook. Knowledge, Kyiv (2005) (in Ukrainian)
8. Morze, N. V.: *Osnovy Iformatciyno-Comynikacynyh Tehnologiy*. Publishing Group, Kyiv (2006) (in Ukrainian)
9. Oliynyk, V. V.: *Distantciyna Osvita za Kordonom ta v Ukraine: Styslii Analitychniy Oglyad*. CIPPO, Kyiv (2010) (in Ukrainian)
10. Orel, A. M.: *Tutor s Raznyh Tochek Zreniya*. http://www.ou-link.ru/654/bulletin_654_8/tutor-3.htm (in Russian)
11. Starov, M. I., Chvanova, M. S., Vislobokova M. V.: *Psihologo-Pedagogicheskie Problemy pri Distantcionnom Obuchenii*. *Distance education*, 12(2), 26–30 (2012) (in Russian).
12. Ragan, L. C.: *Good Teaching is Good Teaching: an Emerging Set of Guiding Principles and Practices for the Design and Development of Distance Education*. *CAUSE/EFFECT journal*, 22(1), <http://eee.educuse.edu/ir/library/html/cem9915.html> (1999)
13. McKenzie, B., Mims, N., Bennett, E., Waugh, M.: *Needs, Concerns and Practices of Online Instructors*. *Online Journal of Distance Learning Administration*, 3(3) (2011)

1.5 ICTERI Tutorials

UML Profile for MARTE: Time Model and CCSL

Frédéric Mallet¹

Université Nice Sophia Antipolis, Aoste team INRIA/I3S, Sophia Antipolis, France

`Frederic.Mallet@unice.fr`

Abstract. This 90 minutes tutorial gives a basic introduction to the UML Profile for MARTE (Modeling and Analysis of Real-Time and Embedded systems) adopted by the Object Management Group. After a brief introduction to the UML profiling mechanism, we give a broad overview of the MARTE Profile. Then, the tutorial shall focus on the time model of MARTE and its companion language CCSL (Clock Constraint Specification Language).

Keywords. UML Profile, Real-Time, Embedded systems, MARTE, CCSL

Key terms. StandardizationProcess, UbiquitousComputing, ConcurrentComputation, ModelBasedSoftwareDevelopmentMethodology

1 Audience and focus

The targeted audience is academics or industrials interested in high-level modeling with UML and its application to the analysis of real-time and embedded systems.

The tutorial does not require any preliminary background as it will give a high-level and broad description of the UML Profile as well as a closer focus on its time model. To ensure that a large public can follow the presentation, we should start by a brief overview of UML light-weight extension mechanism, the so-called profiling mechanism.

2 Topic

The UML profile for Modeling and Analysis of Real-Time and Embedded systems, referred to as MARTE [1], has been adopted by the OMG in November 2009 and revised in June 2011. It extends the Unified Modeling Language (UML) [2] with concepts required to model embedded systems.

This ninety minutes tutorial gives a basic introduction to the UML Profile for MARTE. After a broad view of the Profile, the tutorial shall focus on the time model of MARTE and its companion language CCSL (Clock Constraint Specification Language).

2.1 General Introduction to MARTE

Figure 1 shows the general structure of MARTE.

The General Component Modeling (GCM) and Repetitive Structure Modeling (RSM) packages offer a support to capture the application functionality. GCM defines basic concepts such as data flow ports, components and connectors. RSM provides concepts for expressing repetitive structures and regular interconnections. It is essential for the expression of parallelism, in both application modeling and execution platform modeling; and for the allocation of applications onto execution platforms.

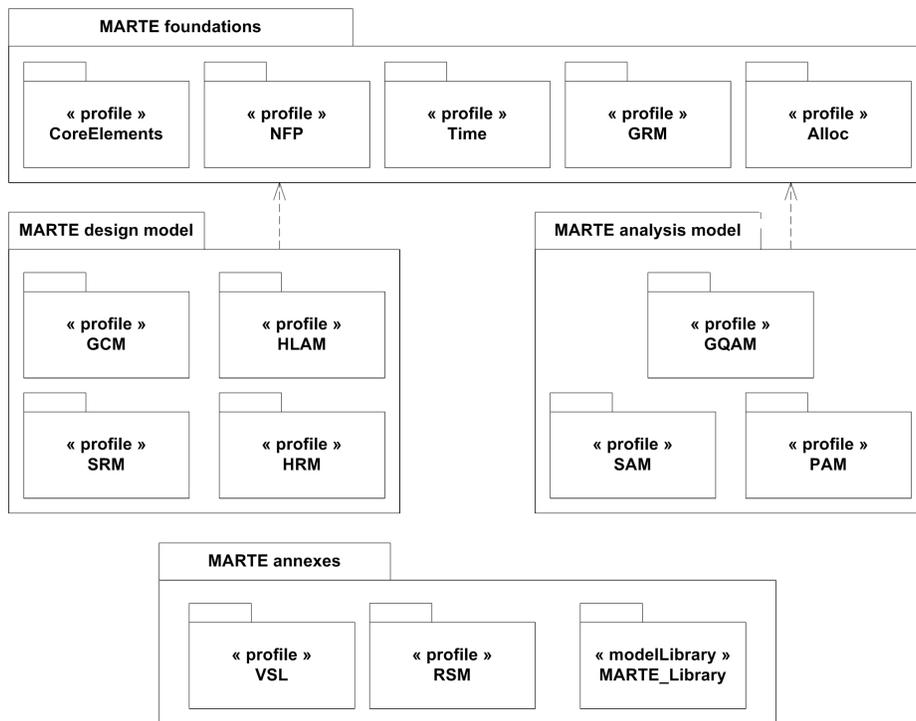


Fig. 1. Structure of MARTE specification

The Hardware Resource Modeling (HRM) package, which specializes the concepts of GCM into hardware devices such as processor, memory or buses, allows the modeling of the execution platforms in MARTE. The Allocation (Alloc) package allows the modeling of the space-time allocation of application functionality on an execution platform. Both the HRM and Alloc packages can be used with the RSM package for a compact modeling of repetitive hardware (e.g., grids of processing elements) and data and computation distributions of a parallel application onto such a repetitive hardware.

The models described with the previous packages can be refined with temporal properties specified within the Time package [3]. Such properties are typically clock constraints denoting some activation rate constraints about considered components. The concepts of the Time package are often used with the Clock Constraint Specification Language (CCSL) [4, 5], which was initially introduced as a non-normative annex of MARTE.

2.2 MARTE Time Model

In MARTE, a *clock* c is a totally ordered set of *instants*, \mathcal{I}_c . In the following, i and j are instants. A *time structure* is a set of clocks \mathcal{C} and a set of relations on instants $\mathcal{I} = \bigcup_{c \in \mathcal{C}} \mathcal{I}_c$. CCSL considers two kinds of relations: *causal* and *temporal* ones. The basic causal relation is *causality/dependency*, a binary relation on \mathcal{I} : $\preceq_C \subset \mathcal{I} \times \mathcal{I}$. $i \preceq j$ means i causes j or j depends on i . \preceq is a pre-order on \mathcal{I} , *i.e.*, it is reflexive and transitive. The basic temporal relations are *precedence* (\prec), *coincidence* (\equiv), and *exclusion* ($\#$), three binary relations on \mathcal{I} . For any pair of instants $(i, j) \in \mathcal{I} \times \mathcal{I}$ in a time structure, $i \prec j$ means that the only acceptable execution traces are those where i occurs strictly before j (i precedes j). \prec is transitive and asymmetric (reflexive and antisymmetric). $i \equiv j$ imposes instants i and j to be coincident, *i.e.*, they must occur at the same execution step, both of them or none of them. \equiv is an equivalence relation, *i.e.*, it is reflexive, symmetric and transitive. $i \# j$ forbids the coincidence of the two instants, *i.e.*, they cannot occur at the same execution step. $\#$ is irreflexive and symmetric. A consistency rule is enforced between causal and temporal relations. $i \preceq j$ can be refined either as $i \prec j$ or $i \equiv j$, but j can never precede i .

In this paper, we consider discrete sets of instants only, so that the instants of a clock can be indexed by natural numbers. For a clock $c \in \mathcal{C}$, and for any $k \in \mathbb{N}^*$, $c[k]$ denotes the k^{th} instant of c .

Specifying a full time structure using only instant relations is not realistic since clocks are usually infinite sets of instants. Thus, an enumerative specification of instant relations is forbidden. The Clock Constraint Specification Language (CCSL) defines a set of time patterns between clocks that apply to infinitely many instant relations [4].

The UML Profile for MARTE proposes several specific stereotypes in the Time chapter to capture CCSL specifications. Figure 2 briefly describes the three main stereotypes. Boxes with the annotation «metaclass» denote the UML concepts on which our profile relies, so-called metaclasses. Boxes with stereotype are the concepts introduced by MARTE, *i.e.*, the stereotypes. Arrows with a filled head represent extensions, whereas normal arrows indicate properties of the introduced stereotypes. **Clock** extends UML Events to spot those events that can be used as time bases to express temporal or logical properties. **ClockConstraint** extends UML Constraints to make an explicit reference to the constrained clocks. **TimedProcessing** extends **Action** to make explicit their start and finishing events. When those events are clocks, then a **ClockConstraint** can constrain the underlying action to start or finish its execution as defined in a CCSL specification.

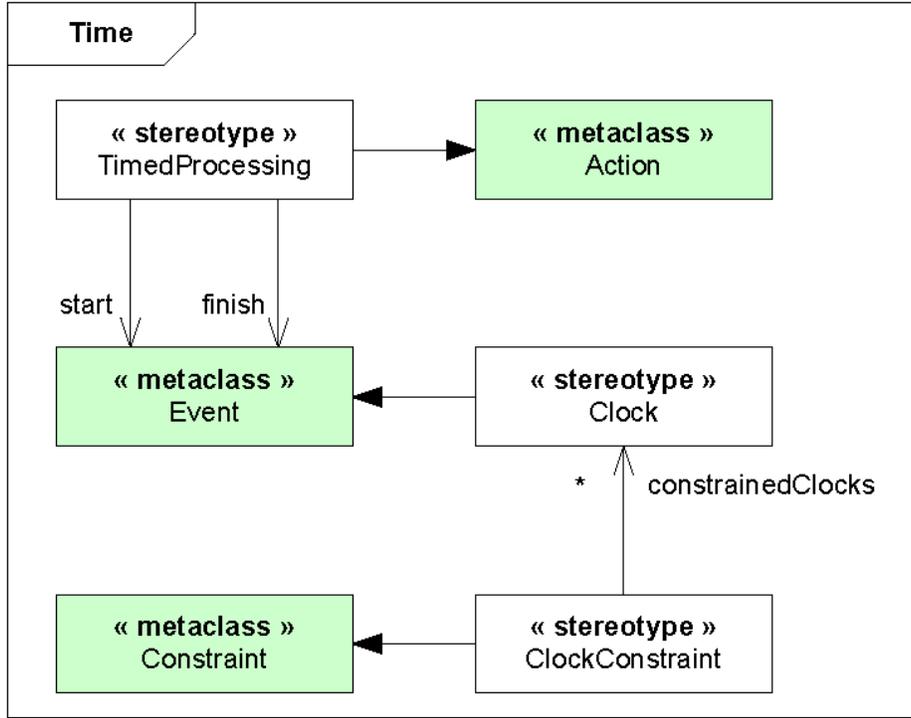


Fig. 2. Excerpt of the MARTE Time Profile

2.3 The Clock Constraint Specification Language

On top of MARTE clocks, the Clock Constraint Specification Language defines a set of operators (relations and expressions) [4]. As an example, consider the clock relation *precedence* (denoted \prec), a transitive asymmetric binary relation on \mathcal{C} : $\prec \subset \mathcal{C} \times \mathcal{C}$. If *left* and *right* are two clocks, $left \prec right$, read 'left precedes right', specifies that the k^{th} instant of clock *left* precedes the k^{th} instant of clock *right*, for all k . More formally: For a pair of clocks $(left, right) \in \mathcal{C} \times \mathcal{C}$, $left \prec right$ means $\forall k \in \mathbb{N}^*, left[k] \prec right[k]$. Similarly, let us consider the transitive and reflexive binary relation on \mathcal{C} called *isSubclockOf* and denoted \sqsubset . $left \sqsubset right$ (read *left* is a sub clock of *right*) means that for all k , the instant $left[k]$ of *left* coincides with exactly one instant of *right*. More formally: $left \sqsubset right$ means $\forall k \in \mathbb{N}^*, \exists n \in \mathbb{N}^* | left[k] \equiv right[n]$. The relation \sqsubset is order-preserving. All the coincidence-based relations are based on *isSubclockOf*. When both $left \sqsubset right$ and $right \sqsubset left$ then we say that *left* and *right* are synchronous: $left \equiv right$.

A CCSL specification consists of clock declarations and conjunctions of *clock relations* between *clock expressions*. A clock expression defines a set of new clocks from existing ones, most expressions deterministically define one single clock. An

example of clock expression is *delay* (denoted $\$: \mathcal{C} \times \mathbb{N}^* \rightarrow \mathcal{C}$). $c \$ n$ specifies that a new clock is created and is the exact image of c delayed for n instants: $o = c \$ n$ defines a clock $o \in \mathcal{C}$ such that $\forall k \in \mathbb{N}^*, o[k] \equiv c[k + n]$.

By combining primitive relations and expressions, we derive a very useful clock relation that denotes a bounded precedence. $left \boxed{\prec_n} right$ is equivalent to the conjunction of $left \boxed{\prec} right$ and $right \boxed{\prec} (left \$ n)$. The special case, when n is equal to 1 is called *alternation* and is denoted $left \boxed{\sim} right$ (reads *left alternates with right*).

3 Conclusion

The UML Profile for MARTE is dedicated to the modeling and analysis of real-time and embedded systems. Its time model relies on a notion of clock borrowed from the synchronous languages [6] and their polychronous extensions [7]. Those clocks can be logical or physical. The time model also provides a support to build causal and temporal constraints to force the clocks to tick according to predefined patterns. Thus, the evolution of the clocks imposes an execution semantics on the underlying UML MARTE model. Whereas the MARTE time model provides the notions of clocks and constraints, its companion language, the Clock Constraint Specification Language provides a syntax to define the constraints themselves. This brief tutorial introduces the main concepts of MARTE time model and gives an overview of CCSL.

Acknowledgments

This work has been partially funded by the PRESTO Project (ARTEMIS-2010-1-269362).

References

1. OMG: UML Profile for MARTE, v1.1. Object Management Group. (June 2011) formal/2011-06-02.
2. OMG: UML Superstructure, v2.4.1. Object Management Group. (August 2011) formal/2011-08-06.
3. André, C., Mallet, F., de Simone, R.: Modeling time(s). In: 10th Int. Conf. on Model Driven Engineering Languages and Systems (MODELS '07). Number 4735 in LNCS, Nashville, TN, USA, ACM-IEEE, Springer (September 2007) 559–573
4. André, C.: Syntax and semantics of the Clock Constraint Specification Language (CCSL). Research Report 6925, INRIA (May 2009)
5. Mallet, F., André, C., de Simone, R.: CCSL: specifying clock constraints with UML/Marte. *Innovations in Systems and Software Engineering* 4(3) (2008) 309–314
6. Benveniste, A., Caspi, P., Edwards, S.A., Halbawachs, N., Le Guernic, P., de Simone, R.: The synchronous languages 12 years later. *Proc. of the IEEE* 91(1) (2003) 64–83

7. Le Guernic, P., Talpin, J.P., Le Lann, J.C.: Polychrony for system design. *Journal of Circuits, Systems, and Computers* **12**(3) (2003) 261–304
8. Mallet, F.: Logical Time in Model Driven Engineering. *Habilitation à diriger des recherches*, Université Nice Sophia-Antipolis (November 2010)

Biography

Frédéric Mallet is an associate professor at Université Nice Sophia Antipolis. He is a permanent member of the Aoste team-project, a joint team between INRIA and I3S laboratory. He received a PhD in Computer Science in 2000 and his habilitation degree [8] in 2010. Since 2007, he has been heavily involved in the definition, finalization and revision of the UML Profile for MARTE¹ and has been a voting member of the successive finalization and revision task forces for MARTE at the OMG. His main research interests include the definition of models for the specification of functional and non-functional properties of real-time and embedded systems. He also develops tools and techniques for the validation and verification of such models.

¹ <http://www.omgmarte.org>

Ontology Alignment and Applications in 90 Minutes

Vadim Ermolayev¹ and Maxim Davidovsky¹

¹ Department of IT, Zaporozhye National University
Zhukovskogo st. 66, 69063 Zaporozhye, Ukraine

vadim@ermolayev.com, m.davidovsky@gmail.com

Abstract. In this paper, we describe the structure and outline the content of a short tutorial on Ontology Alignment. The tutorial is planned in three parts within an overall timeframe of 90 minutes. Part 1 covers the fundamentals of ontology alignment and offers basic definitions, problem statements and problem classification based on the span, dynamics, direction, and distribution settings. This material is illustrated by: (i) using a walkthrough example of two elementary ontologies in Bibliographics domain; and (ii) offering a deeper discussion of one of the exemplar problems of ontology alignment – ontology instance migration – which has a practical utility for real world applications. The second part presents a software solution for ontology instance migration problem. The solution is demonstrated on the pair of Bibliographic ontologies of our walkthrough example. Part 3 puts ontology alignment in the context of several categories of applications which are important for the industries and the knowledge economy as a whole. The applications of ontology alignment in those categories are overviewed and requirements to the solutions are extracted.

Keywords. Ontology, ontology alignment, knowledge-based application, agent, argumentation, negotiation, information flow, ontology instance migration

Key terms. KnowledgeRepresentation, KnowledgeManagementMethodology, KnowledgeManagementProcess, KnowledgeTechnology, ICTTool

1 Introduction

This paper outlines the tutorial on the basics and problems of Ontology Alignment. The material is illustrated by our agent-based solution for ontology instance migration problem – one of practically important sub-problems in ontology alignment. The demand for applications of ontology alignment in real world applications is also presented. The tutorial, though given for the first time, is based in parts on our previous tutorial on Agent-Based Ontology Alignment [1]. This tutorial differs from [1] in the following: (i) it is broader in scope as covers not only agent-based approaches to align ontologies; and (ii) it is more oriented to reviewing industrial applications of ontology alignment and analyzing their requirements to the technology.

1.1 Structure and Timeframe

Part 1 targets a broad audience of those who are interested in the problems of ontology alignment in general and starts at a relatively basic level. It begins with informal definition of ontology alignment and puts the problem into the context of the other knowledge harmonization and integration problems. It further explains the motivation to study the methods of ontology alignment. Further the basic formalisms for ontology alignment are introduced and explained using an incremental approach. The generic ontology alignment problem is stated first and illustrated by the walkthrough example. This generic problem statement is further refined by offering a classification of the types of ontology alignment problems. A particular attention is paid to the ontology instance migration problem as a sub-problem of ontology alignment. The time frame for the first part of the tutorial is 30 minutes¹. A standard configuration of presentation equipment is required: 1 beamer, 1 presentation screen, 1 microphone for the presenter, 1 additional microphone for the questions from the audience.

Part 2 offers a more practical material as it is focused on the presentation of the agent-based software solution for the ontology instance migration problem. The material of this part covers the presentation of the: (i) solution architecture; (ii) methodology shaping out the workflow; (iii) software demonstration that migrates instances from one to the other ontology of our walkthrough example. The time frame for the second part of the tutorial is also 30 minutes. Part 2 uses two independent presentation channels: one for the tutor and the other for software demonstration. Therefore it requires an enhanced configuration of presentation equipment: 2 beamers, 2 presentation screens, 2 microphones for the presenters, 1 additional microphone for the questions from the audience.

Part 3 is focused on the discussion of the importance of ontology alignment technology for real world applications. It starts with revisiting the motives to have this technology in place and proves the necessity of having the solutions for several categories of ICT applications, particularly in information and knowledge processing. In fact a review of applications, their specific requirements, and available solutions is given in this concluding part of the tutorial to provide a holistic, cross-domain view on the role of ontology alignment as a fundamental technology for today's knowledge economy. Similarly to parts 1 and 2, the time frame for part 3 of the tutorial is 30 minutes. Similarly to part 1, part 3 requires a standard set of presentation equipment.

The whole tutorial is therefore given in 90 minutes. A small break could be planned after Part 2 if the audience wishes to do so for having some discussions or posing in-depth questions. Though questions are allowed to be posed at any time, all three parts are planned with 5-minute question and answer sessions at their ends.

¹ Timings are given approximately. Small deviations could occur depending on the number of questions coming from the audience.

1.2 A Walkthrough Problem and Example

An example problem of ontology alignment that is used throughout the tutorial for detailed discussions is the **ontology instance migration** problem. The problem statement for ontology instance migration is presented in Section 2. The approach and software for solving this problem is demonstrated in Section 3. The applications that require the migration of ontology instances are mentioned among those discussed in Section 4.

Besides that, a very simple and artificial example of two different `Biblio` ontologies is used for illustrations throughout the tutorial. The structural schemas and assertional parts of these ontologies are provided in the support material at http://isrg.kit.znu.edu.ua/a-boa/index.php/A-BOA_Walkthrough_Problem_and_Example.

1.3 Support Materials, Discussions, and Contributions

For additional support materials a reader is advised to visit the A-BOA Wiki (isrg.kit.znu.edu.ua/A-BOA/) which has been developed for our previous tutorial on Agent-Based Ontology Alignment [1]. A-BOA Wiki is a Semantic MediaWiki based collaborative platform and a resource providing teaching content and discussion functionality.

1.4 Motivation to Study Ontology Alignment

The world around us is multi-faceted and polysemic in a sense that a model of the world developed in the mind of an individual or by a social group may be different from the model of the others. Knowledge-based systems reflect this fact in their knowledge representations. However, we do many things across several facets or even across subject domains. So, the knowledge representations of the corresponding facets of knowledge representation have to be brought into a harmonized or aligned state to enable proper communication, coordination or information processing.

`Biblio` ontologies give a simple example of such different facets, or knowledge representations, for the same body of knowledge about conference papers. Imagine that `Biblio-2` is the knowledge representation of a conference management system, while `Biblio-1` is the model for a paper repository used by a publisher for book production. The descriptions of the papers that have been accepted for a conference have to appear in the publisher's paper repository. Similarly, the publisher's information about the page limits has to be given to the conference management system to instruct the authors at proper time. Knowledge representations of `Biblio-1` and `Biblio-2` have therefore to be aligned for enabling seamless transformation and transfer of individual records between these two distributed knowledge-based systems. The tutorial will teach how such alignments could be done and what the complications in that activity are.

An attendee will learn that an alignment is essentially a result of applying a set of formal transformations to a knowledge representation – to its structure and individual assertions. An alignment allows interpreting knowledge that is external to the inter-

preter in the same way it interprets its own knowledge schema and assertions. For example, if an alignment of `Biblio-2` to `Biblio-1` exists, the publisher, who is the owner of `Biblio-1` may seamlessly import the assertions about the accepted papers to its production repository. Similarly, an alignment of `Biblio-1` to `Biblio-2` is required by conference organizers to get the publisher's information about publication constraints like page limits.

In a summary, ontology alignment has to be a technology at hand for all those who develop distributed constellations of knowledge-based systems that require collaboration across the nodes. Building ontology alignments efficiently and effectively is also important for the management and maintenance of such systems. Indeed, the fact that you have developed a perfect ontology alignment for your system does not yet allow you to retire. World changes and these changes are reflected in some facets of knowledge representations sporadically and without informing the other nodes. Hence the alignment activity has to be repeated in order to bring the whole system to a harmonized state.

2 Basics and Problems of Ontology Alignment

This section of the tutorial presents the formal problem statement and classification of ontology alignment problems, discusses one of the problem statements – for the ontology instance migration problem in more detail.

Following Euzenat and Shvaiko [2], an **ontology** is formally denoted as a tuple $O = \langle C, P, I, T, V, \leq, \perp, \in, = \rangle$ where C is the set of *concepts* (or *classes*); P is the set of *properties* (*object* and *datatype properties*); I is the set of *individuals* (or *instances*); T is the set of *datatypes*; V is the set of *values*; \leq is a *reflexive, anti-symmetric and transitive relation* on $(C \times C) \cup (P \times P) \cup (T \times T)$ called *specialization*, that form partial orders on C and P called *concept hierarchy* and *property hierarchy* respectively; \perp is an *irreflexive and symmetric relation* on $(C \times C) \cup (P \times P) \cup (T \times T)$ called *exclusion*; \in is a relation over $(I \times C) \cup (V \times P)$ called *instantiation*; $=$ is a relation over $I \times P \times (I \cup V)$ called *assignment*; (the sets C, P, I, T, V are pairwise disjoint). It is also assumed (c.f. [3]), that an ontology O comprises its schema S and the assertional part A (see also Fig. 2):

$$O = \langle S, A \rangle; S = \langle C, P, T \rangle; A = \langle I, V \rangle \quad (1)$$

Ontology schema is also referred to as a **terminological component** (TBox). It contains the statements describing the concepts of O , the properties of those concepts, and the axioms over the schema constituents. The **set of individuals**, also referred to as an **assertional component** (ABox), is the set of the ground statements about the individuals and their attribution to the schema – i.e. where these individuals belong.

Ontology matching is denoted as a process of discovering the correspondences (or *mappings*) between the elements of different ontologies. A *mapping* (or a *mapping rule* [2]) is a tuple $m = \langle e, e', \mathfrak{R}, n \rangle$, where: e, e' are the elements of C, R, I, T, V of the

respective ontologies O and O' ; $\mathfrak{R} = \{c, \subseteq, \equiv, \supseteq, \supseteq\}$ is a set of relations; and n is a confidence value (typically in the range of $[0,1]$).

Finally, **ontology alignment** is denoted as the result of applying the discovered set of mapping rules to the respective ontologies. A generic ontology matching process and ontology alignment are described and pictured in more detail at http://isrg.kit.znu.edu.ua/a-boa/index.php/Basic_Definitions_and_Generic_Problem_Statement.

Based on the features of participating ontologies and the span of e, e' across C, P, I, T, V -s of O and O' a classification of the problems of finding ontology alignments could be outlined and formally stated. Graphical interpretation of some of these problems is described in more detail at http://isrg.kit.znu.edu.ua/a-boa/index.php/Classification_of_Ontology_Alignment_Problems. The dimensions along which the problems are classified are:

Complete (C), structural (S), or assertional (A) alignment

Static (S) versus dynamic (D) aligned ontologies

Bi-directional (B) versus uni-directional (U) alignment

Fully distributed (D) settings versus the presence of a central (C) referee ontology

A generic ontology alignment process may therefore be classified as a complete static bi-directional alignment using central referee ontology (CSBC). Our walk-through problem of ontology instance migration could be classified as assertional, static, uni-directional, distributed (ASUD) ontology alignment problem.

Yet another important feature for classifying ontology alignment processes is the presence of iterations for the refinement of alignments. All the processes discussed above are one-shot. However, the resulting alignments may appear to be of insufficient quality after their evaluation. Iterative ontology alignment processes aim at improving this shortcoming by incorporating the evaluation step and the refinement cycle in the process – please refer to (http://isrg.kit.znu.edu.ua/a-boa/index.php/Classification_of_Ontology_Alignment_Problems) for a graphical illustration. Iterative ontology instance migration process is discussed in more detail below. Our agent-based software prototype toolset for solving this problem is presented in Section 3.

One of the practically important ontology alignment problems, especially in fully distributed and dynamic settings, is the problem of transferring the individuals of one (source) ontology to the empty assertional part of the other (target) ontology [4].

Let us consider two arbitrary ontologies $O^s = (S^s, A^s)$ and $O^t = (S^t, A^t)$ conceptualizing the semantics of the same universe of discourse U – for example O^s and O^t are the two ontologies describing the same subject domain. U could be regarded as a collection of ground facts: $U = \{f\}$. Essentially, O^s and O^t are the interpretations of U . These ontologies would be considered identical if and only if:

$$\forall f \in U \text{ int}_{I^s}(f) \equiv \text{int}_{I^t}(f), \quad (2)$$

where $\text{int}_I(f)$ is the interpretation of the fact f by the individuals from I of ontology O .

Consequently, an abstract metric of interpretation difference $idiff(U, O^s, O^t)$ could be introduced. The value of $idiff$ will be equal to zero for identical ontologies and will increase monotonically to one with the increase of the number of $f \in U$ such that $\neg(\text{int}_{I_s}(f) \equiv \text{int}_{I_t}(f))$. Hence, $idiff = 1$ iff $\forall f \in U \neg(\text{int}_{I_s}(f) \equiv \text{int}_{I_t}(f))$. $(1 - idiff)$ may further be interpreted [4] as balanced F -measure.

Ontologies O^s and O^t are structurally different if their schemas differ: $S^s \neq S^t$. This structural difference may be presented as a transformation $T: S^s \rightarrow S^t$. Transformation T may be sought in the form of the set of nested transformation rules over the constituents of S^s resulting in the corresponding constituents of S^t .

Let us assume now that, given two structurally different ontologies O^s and O^t , the ABox of O^s contains individuals ($I^s \neq \emptyset$), while the ABox of O^t is empty ($I^t = \emptyset$). The problem of minimizing $idiff(U, O^s, O^t)$ by: (i) taking the individuals from I^s ; (ii) transforming them correspondingly to the structural difference between O^s and O^t using T ; and (iii) adding them to I^t – is denoted as **ontology instance migration problem**.

Theoretically ontology instance migration problem can be solved in one shot. In practice however each of the sub-tasks (ii-iii) may result in the loss of assertions [4]. Therefore an iterative refinement of the solution could yield results with a lower resulting $idiff$ value. Hence, the problem has to be solved using an iterative ontology alignment process. Essentially, an iterative solution of ontology instance migration problem develops a sequence of O^s states $O_{st_i}^s$ in a way to minimize the $idiff(U, O^s, O^t)$ in a way that:

$$idiff(U, O_{st_i}^s, O^t) < idiff(U, O_{st_j}^s, O^t) \rightarrow i > j, \quad (3)$$

where: $O_{st_i}^s$ is O^s in the state after accomplishing iteration i ; i, j are iteration numbers.

3 A Solution for Ontology Instance Migration Problem

This section demonstrates our agent-based solution for the ASUD ontology alignment problem stated above as **ontology instance migration problem**. This problem has been chosen as it possesses significant practical interest in real world applications, in particular for Ontology Engineering and Management in distributed and dynamic settings [4]. Instance migration in our solution is performed iteratively, so the alignment is refined from iteration to iteration.

Many influential publications, for example [5], envision that intelligent software components, like agents, need to be used together with ontologies for making semantic technologies accepted and effective in open and decentralized scenarios. For such agent based solutions, comprising industrial applications, the heterogeneity problem is the challenge that has to be faced. Ontology alignments are a means to solve the chal-

lence. From the other hand agents, being the recipients of ontology alignment solutions, may help solving ontology alignment problems.

For a graphical illustration and more details of a simplified agent-based architecture for solving a generic ontology alignment problem please refer to http://isrg.kit.znu.edu.ua/a-boa/index.php/Theoretical_Foundations_and_Demonstration. The architecture introduces the wrapper agents W and W' for ontologies O and O' respectively. Agent R wraps the central referee ontology O' and helps W and W' finding the proper mappings using O' (a matchmaker function). W and W' produce their own sets of mappings M and M' in collaboration with each other (a fully distributed problem setting) or also in collaboration with R (the problem setting with a central referee ontology). At the Apply Mappings step M and M' are autonomously applied by W and W' to O and O' . A problem in developing such an agent-based solution is how do the agents collaborate and develop these mappings.

The presented solution is based on automated meaning negotiations between agents [6] as a way to discover structural differences between the schemas of O and O' . Similarly to [7], this approach aims at aligning ontologies by parts (*contexts*) that are relevant to a particular negotiation encounter. Negotiations imply iterative monotonic reduction of *semantic distances* between the contexts. An agent uses *propositional substitutions* which may reduce the distance and support them with *argumentation*. The process is stopped when the distance reaches a commonly accepted threshold or the involved parties exhaust their propositions and arguments. As opposed to the Argumentation Framework based approaches, this approach addresses the entire process of semantic reconciliation between ontologies and does not require off-the-shelf mappings.

The methodology used in our solution comprises several steps in the workflow. Steps (I) and (II) correspond to Discover Mappings, step (III) is for Applying Mappings, step (IV) corresponds to the step of evaluation and making decision about undertaking one more iteration. Iteration loop however does not involve mappings discovery in our solution. Instead, the mappings are revised manually by a knowledge engineer based on the list of migration failures in the migration log. Step (V), though important in practice, is not demonstrated.

Biblio-1 and Biblio-2 are used as examples of O and O' . The demonstrated agent-based solution is evaluated by comparing to our former work [4] where Ontology Difference Visualizer (ODV) tool [8] was used for discovering the structural difference between aligned ontologies.

Ontology instance migration process starts with the step (I) of discovering the structural difference between O and O' . Only TBoxes of the ontologies are used as the sources. Structural difference is discovered by the SDiff Discovery Engine (SDDE) [9] – a system of collaborative software agents negotiating on semantic contexts [10] for finding mappings $M': S \rightarrow S'$. For demonstration purposes discovered structural difference is visualized using UML extension [8]. The mappings are further written down by SDDE as instance transformation rules [4] at the subsequent step (II). Instance Migration Engine (IME) is invoked at step (III) to perform the instance transformations according to these transformation rules. All the cases in which IME fails

to perform the transformation are recorded to the instance migration log. Step (IV) involves a knowledge engineer who checks the migration log and decides if a refinement is required. If so, he starts the new iteration by refining the set of the transformation rules based on his analysis of the failure cases and using the rule editor of IMS at step (II). The refined set of rules is fed to the IMS at step (III). The loop continues until the knowledge engineer decides that further refinement is not possible, or all the instances of I^s are migrated to I^t .

4 Applications of Ontology Alignment

In this part of the tutorial a few selected categories of applications that require aligning information or knowledge representations are analyzed. A broader spectrum of applications is surveyed in [11]. In particular, attention is paid to the requirements related to ontology alignments that are posed by the applications in each category. A particular ontology alignment problem fitting to these requirements is also outlined.

A good survey of ontology-based applications is [12]. Ontology matching and alignment applications are discussed in [2]. Another comprehensive summary of ontology matching techniques and applications is [13]. In addition to these surveys, the publications surveying or reporting ontology alignment approaches are for example Chuttur [14], Vázquez-Naya et al. [15], Zhdanova et al. [16], Euzenat et al. [17]. Based on these inputs the following several typical application categories are analyzed in the tutorial with a focus on real world applications.

4.1 Distributed Information Retrieval

Distributed Information Retrieval (DIR) is an important category of applications that assist retrieving and fusing information from heterogeneous, distributed, and independent information resources. Ontologies in DIR are used for representing the structures of information at different nodes and for translating or transforming user queries and system responses. In particular, ontologies in DIR are important for extracting information or knowledge satisfying the semantics and the context of a user query. Ontology alignments are required:

- At query transformation step – for correlating query structure and semantics with different information resource schemas and metadata and building respective partial queries
- At query result fusion step – transforming and putting together the retrieved information instances

Hence, a solution of an SSUD ontology alignment problem is required for query transformation and of an ASUD problem for results fusion and delivery to a user. A **critical requirement** at the latter step is **high recall** as it is important not to miss any potentially relevant information while irrelevant individuals can be filtered out using other techniques. One more important requirement to an ontology alignment solution in DIR is its **scalability** in terms of the complexity and number of aligned ontologies.

4.2 Human-Machine Dialogues

Ontology alignments are used in human-machine interaction for providing mutual understanding between a user and a processing node. A software agent may represent a processing node in such interactions as an intelligent wrapper. Ontologies and their alignments can be used to obtain a formalizable set of requirements, structures, queries, etc. from informal or poorly structured user descriptions. As a rule such dialogs are run in iterative way. Hence, **iterative ontology alignment** methods fit to this category of applications better.

Brasoveanu et al. [18] argue the importance of using generic multimodal ontologies on the Semantic Web and propose an approach to enhance human-agent interaction based on multimodal ontologies. Guzzoni et al. [19] propose a toolkit-based approach for modeling human-agent interaction. Their toolset provides a means to model different aspects of an intelligent assistant such as: ontology-based knowledge structures; service-based primitive actions; composite processes and procedures; natural language and dialog structures. Tijerino et al. [20] report a framework for human-agent collaboration for the purpose of problem solving on the Semantic Web. In human-machine dialogue scenarios the most critical requirements are **adaptability**, **integrativity**, and **scalability** that allow enhancing human-machine mutual understanding.

4.3 Ontology Evolution, Versioning, Refinement

Ontology evolution, versioning, and refinement are important problems in Ontology Engineering (OE) and Management (KM) applications. Solutions are required for adequately representing knowledge in changing domains. Ontology alignment is one of the enabling technologies in these applications. Indeed, all three problems cope with transforming a source ontology revision to a target state (revision) that fits to the requirements causing the transition. Important aspects of this transition are that the target revision has to: (i) be consistent; (ii) re-use the source as much as allowed by the requirement of being consistent

Ontology alignments are used both to ensure consistency and maximal possible degree of re-use. Provided that the source revision is consistent, for proving that the resulting ontology revision is consistent it is sufficient to build the complete static bi-directional alignment (CSBC or CSBD problems). For the proper re-use of the source revision the solution of a uni-directional alignment problem will fit. For example a typical sub-task in an ontology refinement process is ontology instance migration from the source revision to the target revision [4]. A balanced **combination of** appropriately high **recall and precision** is an essential requirement for the instance migration solution.

4.4 Service Composition

The automation of web service composition or orchestration at run time is a challenging problem in Service Science which is intensively researched in the last decade. The

complexity of the problem is caused by the inherent distributed character of software systems based on the use of services (for example Web services), the openness of these systems, and the dynamic character of their configurations and constellations. A sub-stream of research in the field develops the frameworks for services that intensively use ontologies as service descriptions – Semantic Web Services. Two prominent examples of these frameworks are OWL-S [21] and WSMO/L/X [22] which however do not fully solve runtime service composition problem. More advanced approaches exploit collaborative agents as service wrappers for managing services and service brokers or mediators for manipulating their descriptions in a runtime composition process (for example [23]). Like in Ontology Engineering and Management, a balanced **combination of** appropriately high **recall and precision** is an essential requirement for service composition. The **scalability** of the solution is also important.

The aspects of ontology reconciliation with respect to Web services and their composition are elaborated in [24, 25, 26]. An important requirement for such systems is the capability of adaptation and integration for providing compliant access and making the use of aggregate and atomic services more convenient.

5 Learning Outcomes

By the end of the tutorial the participants will:

- Learn the basics of ontology alignment that will enable them to understand the notions of an ontology, ontology mapping, the process of ontology matching, and the alignment as a result of matching process
- Learn the generic ontology alignment problem and the classification of its flavors based on the features of distributedness, the span of alignment, the direction of alignment, and the dynamic character of the source ontologies. Specifically, learn about the ontology instance migration problem as one of the ontology alignment problems.
- Be able to differentiate between one-shot and iterative ontology alignment methods and judge about the appropriateness of using this or that kind of a method in a particular setting
- Learn about one of the agent-based solutions for ontology alignment (ontology instance migration problem)
- Learn that ontology alignment is a very important, enabling technology for several kinds of the applications of distributed knowledge-based systems. In particular, learn which of the requirements of these applications make ontology alignment a challenging task.

References

1. Ermolayev, V., Davidovsky, M.: Agent-Based Ontology Alignment: Basics, Applications, Theoretical Foundations, and Demonstration. Tutorial Paper. In: Dan Burdescu, D., Akerkar, R., Badica, C. (eds.) Proc. WIMS 2012, 11-22, ACM (2012)

2. Euzenat J., Shvaiko P.: *Ontology Matching*. Berlin Heidelberg, Springer-Verlag (2007)
3. Nardi, D., Brachman, R. J.: An Introduction to Description Logics. In: Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., Patel-Schneider, P. F. (eds.) *The Description Logic Handbook*. Cambridge University Press New York, NY, USA (2007)
4. Davidovsky, M., Ermolayev, V., Tolok, V.: Instance Migration between Ontologies having Structural Differences. *Int. J. on Art. Int. Tools*. 20(6), 1127-1156 (2011)
5. Berners-Lee, T., Hendler, J., Lassila, O.: *The Semantic Web*. *Scientific American*, 284, 28–37 (2001)
6. Ermolayev V., Keberle, N., Matzke, W.-E., Vladimirov, V.: A Strategy for Automated Meaning Negotiation in Distributed Information Retrieval. In: Y. Gil et al. (Eds.): *ISWC 2005 Proc. 4th Int. Semantic Web Conference (ISWC'05)*, 6–10 November, Galway, Ireland. LNCS 3729, pp. 201–215 (2005)
7. Atencia, M., Schorlemmer, M.: Formalising Interaction-Situated Semantic Alignment: The communication product. In: *Proc. of the Tenth International Symposium on Artificial Intelligence and Mathematics (ISAIM'08)*, Fort Lauderdale, Florida, USA (2008)
8. Ermolayev, V., Copylov, A., Keberle, N., Jentsch, E., Matzke, W.-E.: Using Contexts in Ontology Structural Change Analysis. In: Ermolayev, V., Gomez-Perez, J.-M., Haase, P., Warren, P., (eds.) *Proc. CIAO 2010, CEUR-WS*, vol. 626 (2010)
9. Davidovsky, M., Ermolayev, V., Tolok, V.: Agent-Based Implementation for the Discovery of Structural Difference in OWL DL Ontologies. In: Mayr, H. C., Ginige, A., Liddle, S. (eds.) *Proc. 4th Int. United Information Systems Conference (UNISCON 2012)*, LNBP 137, Springer-Verlag, Berlin Heidelberg (2013)
10. Ermolayev, V., Ruiz, C., Tilly, M., Jentsch, E., Gomez-Perez, G. M., Matzke, W.-E.: A Context Model for Knowledge Workers. In: Ermolayev, V., Gomez-Perez, J.-M., Haase, P., Warren, P. (eds.) *Proc. CIAO 2010, CEUR-WS*, vol. 626 (2010)
11. Davidovsky, M., Ermolayev, V., Tolok, V.: A Survey on Agent-Based Ontology Alignment. In: *Proc. 4th Int. Conf. on Agents and Artificial Intelligence ICAART'12*, pp. 355–361 (2012)
12. Gargantilla, J., Gomez-Perez, A.: *OntoWeb: A Survey on Ontology-Based Applications*. Deliverable 1.6. OntoWeb Consortium IST Project IST-2000-29243 (2004)
13. Scharffe, F., Euzenat, J., Le Duc, C., Mocan, A., Shvaiko, P.: *Analysis of Knowledge Transformation and Merging Techniques and Implementations*. KWEB/2004/D2.2.7/0.8 (2007)
14. Chuttur, M. Y.: Challenges Faced by Ontology Matching Techniques: Case Study of the OAEI Datasets. *J. of Information Technology*, 3(1), 33–42 (2011).
15. Vázquez-Naya, J. M., Romero, M. M., Loureiro, J. P., Sierra, A. P.: *Ontology Alignment Overview*. *Encyclopedia of Artificial Intelligence 2009*, pp. 1283–1289 (2009)
16. Zhdanova A., V., de Bruijn, J., Zimmermann, K., Scharffe, F.: *Ontology Alignment Solution*. Deliverable D14 v2.0 (2004)
17. Euzenat, J., Laera, L., Tamma, V., Viollet, A.: *Negotiation and Argumentation Techniques among Agents Complying to Different Ontologies*. Deliverable D2.3.7, KWEB, v1.0 (2006)
18. Brasoveanu, A., Manolescu, A., Spânu, M. N.: *Generic Multimodal Ontologies for Human-Agent Interaction*. *Int. J. of Computers, Communications & Control*, 5(5), 625–633 (2010)
19. Guzzoni, D., Baur, C., Cheyer, A.: *Modeling Human-Agent Interaction with Active Ontologies*. *Artificial Intelligence*, SS-07-04, 52–59 (2007)
20. Tijerino, Y.A., Al-Muhammed, M., Embley, D.W.: *Toward a Flexible Human-Agent Collaboration Framework with Mediating Domain Ontologies for the Semantic Web*. In: *Proc.*

- ISWC 2004 Workshop on Meaning Coordination and Negotiation, Hiroshima, Japan, pp. 131–142 (2004)
21. Martin, D., Paolucci, M., McIlraith, S., Burstein, M., McDermott, D., McGuinness, D., Parsia, B., Payne, T., Sabou, M., Solanki, M., Srinivasan, N., Sycara, K.: Bringing Semantics to Web Services: The OWL-S Approach. In: Cardoso, J., Sheth, A. (eds.) Proc. SWSWPC 2004, LNCS 3387, pp. 26–42 (2004)
 22. Roman, D., Keller, U., Lausen, H., de Bruijn, J., Lara, R., Stollberg, M., Polleres, A., Feiera, C., Bussler, C., Fensel, D.: Web Service Modeling Ontology. *Applied Ontology*, 1(1), 77–106 (2005)
 23. Ermolayev, V., Keberle, N., Kononenko, O., Plaksin, S., Terziyan, V.: Towards a Framework for Agent-Enabled Semantic Web Service Composition. *Int. J. of Web Services Research*, 1(3), 63–87 (2004)
 24. Li, L., Yang, Y.: Agent Negotiation Based Ontology Refinement Process and Mechanisms for Service Applications. *Service Oriented Computing and Applications*, 2, 15–25 (2008)
 25. Paurobally, S., Tamma, V., Wooldridge, M.: A Framework for Web Service Negotiation. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2(4) (2007)
 26. Huang, J., Zavala, R., Mendoza, B., Huhns, M. N.: Reconciling Agent Ontologies for Web Service Applications. In: Proc. of Multiagent System Technologies: Third German Conference (MATES-05). LNAI 3550, pp. 106–117 (2005)

Biographies

Vadim Ermolayev is an associate professor at the Department of Information Technologies (IT) of Zaporozhye National University and the lead of Intelligent Systems Research Group. He is also a research consultant in Semantic Technologies, Intelligent Software Systems, Distributed Artificial Intelligence. The research projects he took part in were focused on: intelligent systems and knowledge representations for enterprises; business and informal process dynamics; intelligent distributed information retrieval; the confluence of agent-based systems and Semantic Web services; ontology engineering, evolution, and refinement; performance management in engineering design. Alignment of knowledge representations was one of important topics in those projects.

Maxim Davidovsky is a PhD candidate at the Department of Mathematical Modeling (MM) of Zaporozhye National University. He also works for the Laboratory of Web-based Technologies and Distance Learning and is the member of Intelligent Systems Research Group at the Department of IT. Maxim received his MSc degree in applied mathematics and accomplished his postgraduate course in mathematical modeling and computational methods at the Department of MM. His research interests are in distributed and decentralized knowledge-based systems and software development. The focus of his current research activity is agent-based ontology alignment and instance migration specifically in distributed and decentralized settings.

Part 2. ICTERI Workshops

**2.1 2nd International Workshop
on Information Technologies
in Economic Research
(ITER 2013)**

Foreword

We would like to offer you the section including a ten papers we have selected for the 2-nd instance of our International Workshop on Information technologies in economic research (ITER 2013) which has been organized as a session in the technical the 9-th International Conference on ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer (ICTERI 2012) held at Kherson, Ukraine on June 19-22, 2013.

The necessity to support decisions by the means of IT at different levels of business process any organization, to verify economical hypotheses and usage of gained knowledge in the learning process requires the use of IT to process relevant information.

Skills of analytical information processing for decision making can effectively be realized only by using information and communication technologies.

The large numbers of economic studies is not supported by modern mathematical framework and ICT, leading to poor quality of the research on both the micro-, macro-and industry levels.

Creation of ITER is intended to familiarize researchers with modern ICT and mathematical methods of information processing in areas such as

- **Business process management:** business process for firms, suppliers, customers, information systems in small and medium business, IT-innovations in management process, R&D company, business intelligence approach, management in virtual organization, e-commerce, cloud technology in business, e-governance.
- **Quantitative methods in economics:** econometrics research on micro- and macro level, business process economical modeling, software package for economic research, modeling industry mergers and acquisitions, finance modeling.
- **IT education for economists:** business informatics curricula, IT-professional training for economical organizations, software programs for economic education, mobile technologies in economical education, e-learning for economists, business games.

Under international level research many scientist and researchers use the Ukrainian and Russian language, which greatly limits the possibilities for the world community to become familiar with published papers. This restricted the number of accepted papers till 10 of the 20 provided for ITER:

Business process management

- Decision Supporting Procedure for Strategic Planning: DEA Implementation for Regional Economy Efficiency Estimation
 - Applying of Fuzzy Logic Modeling for the Assessment of ERP Projects Efficiency
 - Matrix Analogues of the Diffie-Hellman Protocol
 - Binary Quasi Equidistant and Reflected Codes in Mixed Numeration Systems
- Quantitative methods in economics
- Mechanism Design for Foreign Producers of Unique Homogeneity Product
 - Are securities secure: Study of the Influence of the International Debt Securities on the Economic Growth

- How to make high-tech industry highly developed? Effective model of national R&D investment policy
IT education for economists
- Econometric Analysis on the Site “Lesson Pulse”
- Mathematical Model of Banking Firm as Tool for Analysis, Management and Learning
- Features of National Welfare Innovative Potential Parametric Indication’ Information-Analytical Tools System in the Globalization Trends’ Context
Only timely and qualitative preparation of an economic study will provide recommendations and suggestions for decision-makers, to promote the efficient use of material and budgetary resources in organization.

June, 2013

Tanya Payentko
Sergey Kryukov
Vitaliy Kobets

Binary Quasi Equidistant and Reflected Codes in Mixed Numeration Systems

Evgeny Beletsky¹ and Anatoly Beletsky¹

¹Department of Electronics, National Aviation University of Kiev,
1, av. Cosmonaut Komarov, 03680, Kiev, Ukraine

ebeletskiy@gmail.com, abelnau@ukr.net

Abstract. The problem of constructing quasi equidistant and reflected binary Gray code sequences and code in a mixed factorial, Fibonacci and binomial numeration systems is considered in the article. Some combinatorial constructions and machine algorithms synthesis sequences, based on the method of directed enumeration are offered. For selected parameters of sequences all quasi equidistant (for individual cases - reflected) codes with Hamming distance equal to 1 are found.

Keywords. Reflected codes, quasi equidistant sequence, Hamming distance

Key terms. Research, CodingTheory, MathematicalModelling

1 Introduction

Coding theory is one of the most important areas of modern applied mathematics. Beginning of the formation of mathematical coding theory dates back to 1948, when it was published a famous article by Claude Shannon [1]. The growth of codes originally was stimulated by tasks of communication. Later constructed codes found many other applications. Now codes are using to protect data in a computer memory, cryptography, data compression, etc.

The work is devoted to a rather small, but extremely important for applications subset of so-called quasi-equidistant and reflected codes. The class of quasi equidistant codes are sequences of uniform (i.e., containing the same number of bits) of binary code combinations in which any adjacent (neighboring) code sets (words) are at the same Hamming d distance equal to a fixed number of natural numbers (i.e. $d = 1, 2, \dots$) [2]. Equidistant sets include such codes in which any two words (code combinations) are at the same distance d [3].

Finally, we shall refer to the reflected subset quasi equidistant codes with distance $d=1$, the formation of which is based on the principle of mirror reflection? [4]. But if we restrict ourselves to only one mirror, the code sequence will contain the original sequence, after which is the same sequence just re-written in reverse order, which is

unacceptable, since it leads to code repetition. The elimination of repetition can be provided by initial expansion of the number of digits combinations. The essence of the "mirror" reflection of the expansion is explained below as an example of canonical reflected Gray codes and in other sections of this article.

The main objective of this study is to develop algorithms for constructing quasi-equidistant and reflected binary Gray codes as well as code sequences in a mixed factorial, Fibonacci and binomial bases. The method of direct enumeration is the base of algorithms of computer sequences synthesis.

2 Basic of Number System

The history of discrete mathematics and computer science is directly related to the development and introduction of newer principles of representation and encoding digital information, which are based on the *numeration system of numbers*. By a numeration system we understand the way of image sets of numbers using a limited set of characters that form its alphabet, in which the characters (elements of the alphabet) are located in the established order, occupying a certain positions [5]. Any numeration system should be composed of a finite set of non-negative numbers — a range that it encodes. It always includes the number 0 and then follows the natural numbers starting with 1 [6].

There are various numeration system (as well as methods for their classification), whose number is constantly growing. All systems can be divided into the following main classes: positional, not positional and mixed. In the positional numeration systems the same numeric characters (digit) has different meanings in its description depending on the location (level) where it is resides.

By *positional numeration system* is generally understood the p numeration system, which is defined by an integer $p > 1$ — is called a base of numeration system. Unsigned integer N in p numeration system is represented as a finite linear combination of powers of

$$N = \sum_{k=1}^n \alpha_k p^k, \quad (1)$$

where α_k are integers satisfying the inequality $0 \leq \alpha_k \leq (p-1)$, n — the number of digits of the number. The simplest examples of positioning systems (1) can be binary, decimal, and other numeration systems.

In *no positional numeration systems* the value which indicated by the digit does not depend on the position in a number. At the same time the system may impose restrictions on the position of numbers, for example, that they are in descending order. The Roman and many other systems belong to not positional systems.

The *mixed numeration system* is a generalization of the p system, and often refers to the positional numeration systems. The base of mixed numeration system is an increasing sequence of numbers p_k , $k = 1, 2, \dots$, and each N number is presented like linear combination:

$$N = \sum_{k=1}^n \alpha_k p_k,$$

there are some restrictions exist for α_k coefficient.

One of the known examples of the mixed system is a factorial numeration system, in which the bases are the sequence of factorials $p_k = k!$. Another commonly used *Fibonacci* numeration system is a system that is based on Fibonacci numbers. The *Binomial system* in the form in which it is presented in the relevant section of this article, we will also include to a mixed numeration system.

A positive integer is depicted in an arbitrary numeration system as a sequence of symbols $[N] = \alpha_n \alpha_{n-1} \dots \alpha_k \dots \alpha_2 \alpha_1$, where $[N]$ - the number representation in this numeration system, besides each α_k symbol takes r_k bit in general case (if binary alphabet is using).

Note the following general characteristics of quasi equidistant codes with Hamming distance $d = 1$. Let's agree each code sequence starts with zero code. And as result of this agreement the following code after the zero code should be placed with weights 1 and 2, and Further weight codes must alternate *even* (E) — *odd* (O) under the scheme

$$012OEEOE\dots E(O). \tag{2}$$

Scheme (2) is a symbolic form of the tree sequence code combinations. Let's n_e and n_o to be the amount of even and odd code words in a sequence. If the sequence (2) ends up with odd code combination this means $n_e = n_o$, and if even — $n_e = n_o + 1$. This becomes evident:

Statement 1. Inequality

$$0 \leq (n_e - n_o) \leq 1, \tag{3}$$

is a necessary (but not always sufficient) condition for the construction of quasi equidistant codes.

3 Sequences of Gray Codes

Classic Gray codes [7] may be called canonical, since for arbitrary length sequence of combinations are not only quasi equidistant, but also reflected. Let's $G(n)$ – sequence of n-bites classical Gray codes. To construct $(n+1)$ – bites reflected Gray Codes, let's us note as $G_{rc}(n+1)$ – codes, it is just enough to prefix for each source code $G(n)$ the 0 digit and 1 to the left of code group $G^R(n)$ constructed by reflected (reflex or reverse) mirror of $G(n)$ sequence, i.e.

$$G_{rc}(n+1) = 0G(n) \| 1G^R(n) , \tag{4}$$

where $\|$ - is a symbol of concatenation (conjunction of sequences).

According to (4), $G_{rc}(n+1) \equiv G(n+1)$ and as a result sequences of Gray codes of $G(n)$ number of digits $n \geq 2$ are both quasi equidistant and reflected, and besides the line of reflection goes through $2^{n-1} -$ and $(2^{n-1} + 1) -$ code combinations. On the basis of the canonical code $G(n)$, $n \geq 2$, the equidistant Gray codes can be constructed. For example, Tab. 1 show the three 12-bit code quasi equidistant sequences, one of which corresponds to the canonical version of the Gray code.

The first six variants of sequences in the table constructed of canonical option 1 as a result of a variety column rearrangement saving the Hamming distance $d = 1$ of related code combinations. Variants 7-12 are formed as a result of inverse none zero rearrangements of code combinations from appropriate variants 1-6.

Table 1. Three bit quasi equidistant Gray code

Variants of sequence											
1	2	3	4	5	6	7	8	9	10	11	12
000	000	000	000	000	000	000	000	000	000	000	000
001	100	100	001	010	010	100	001	010	010	001	100
011	110	101	101	110	011	101	101	110	011	011	110
010	010	001	100	100	001	111	111	111	111	111	111
110	011	011	110	101	101	110	011	011	110	101	101
111	111	111	111	111	111	010	010	001	100	100	001
101	101	110	011	011	110	011	110	101	101	110	011
100	001	010	010	001	100	001	100	100	001	010	010

The first six variants of sequences in the table constructed of canonical option 1 as a result of a variety column rearrangement saving the Hamming distance $d = 1$ of related code combinations. Variants 7-12 are formed as a result of inverse none zero rearrangements of code combinations from appropriate variants 1-6. As follows from Tab. 1 the only variants 1 (canonical) and 6 of Gray codes belong to a set of three bites reflected codes. At the same time each three bite sequence by (4) statement produce subset of four bite reflected Gray codes. Thereby it is true:

Statement 2. All amounts $L_{or}^{(G)}(n)$ of reflected Gray codes of n number of digits is defined by

$$L_{rc}^{(G)}(n+1) = \begin{cases} n, & \text{if } n \leq 2; \\ 2n!, & \text{if } n \geq 3. \end{cases}$$

3 Factorial Sequence

The integer positive number N in factorial number of numeration system can be represented as

$$N = \sum_{k=1}^n \alpha_k k!, \quad 0 \leq \alpha_k \leq k \tag{5}$$

where $k = 1, 2, \dots, n; \quad 0 \leq \alpha_k \leq k$. Extended form of (5) statement is

$$N = \alpha_n \cdot n! + \alpha_{n-1} \cdot (n-1)! + \dots + \alpha_2 \cdot 2! + \alpha_1 \cdot 1! , \tag{6}$$

Statement (6) is so called numerical, or digital, function [8] of factorial system. There are first 120 decimal numbers (Tab. 2) defined by their α_k coefficients in factorial numeration system.

Table 2. Binary representations of decimal numbers of factorial numeration system

N	$[N_k]_{Fakt}$								
0	0	24	100000	48	1000000	72	1100000	96	10000000
1	1	25	100001	49	1000001	73	1100001	97	10000001
2	10	26	100010	50	1000010	74	1100010	98	10000010
3	11	27	100011	51	1000011	75	1100011	99	10000011
4	100	28	100100	52	1000100	76	1100100	100	10000100
5	101	29	100101	53	1000101	77	1100101	101	10000101
6	1000	30	101000	54	1001000	78	1101000	102	10001000
7	1001	31	101001	55	1001001	79	1101001	103	10001001
8	1010	32	101010	56	1001010	80	1101010	104	10001010
9	1011	33	101011	57	1001011	81	1101011	105	10001011
10	1100	34	101100	58	1001100	82	1101100	106	10001100
11	1101	35	101101	59	1001101	83	1101101	107	10001101
12	10000	36	110000	60	1010000	84	1110000	108	10010000
13	10001	37	110001	61	1010001	85	1110001	109	10010001
14	10010	38	110010	62	1010010	86	1110010	110	10010010
15	10011	39	110011	63	1010011	87	1110011	111	10010011
16	10100	40	110100	64	1010100	88	1110100	112	10010100
17	10101	41	110101	65	1010101	89	1110101	113	10010101
18	11000	42	111000	66	1011000	90	1111000	114	10011000
19	11001	43	111001	67	1011001	91	1111001	115	10011001
20	11010	44	111100	68	1011010	92	1111010	116	10011010
21	11011	45	111011	69	1011011	93	1111011	117	10011011
22	11100	46	111100	70	1011100	94	1111100	118	10011100
23	11101	47	111101	71	1011101	95	1111101	119	10011101

Let's mark $\Phi(k)$ – sequence of n bite factorial codes. In the case where number of digits of code combination from code set $\Phi(k)$ less than k , it is prefixed with required amount of zeros. Let's $\Phi_d(k)$ – sequence of quasi equidistant k – bite factorial codes with Hamming distances among related combinations equal to d . Based on data from Tab. 2 it is easy to create (Tab. 3) sequences $\Phi_1(k)$ for $k = 1$ (singular case), and also $k = 2$ and $k = 3$ created by columns rearrangement of base sequences (variant 1).

Table 3. Sequences of quasi equidistant Factorial Codes

$\Phi_1(k)$								
$k = 1$	$k = 2$		$k = 3$					
1	1	2	1	2	3	4	5	6
0	00	00	000	000	000	000	000	000
1	01	10	010	010	100	100	001	001
	11	11	011	110	101	110	011	101
	10	01	001	100	001	010	010	100
			101	101	011	011	110	110
			100	001	010	001	100	010

Table 3 illustrates one possible method of synthesis of quasi equidistant codes. Its idea is in the following. At the very first stage the base sequence of quasi equidistant codes of n number of digits is created by means of some method (for example, the method of direct search which is examined below). On the second stage a variety of all possible rearrangements of base sequence columns (check out Tab. 3, the correspondent values are of number 1) is done which results in formation of $n!$ different quasi equidistant codes. And finally on the third stage the sequences which contain restricted code combinations are excluded from $n!$ sequences. Such combinations are 110 codes from Tab. 3 highlighted with bold type. So from six three bite sequences the only two generate quasi equidistant factorial sequences. Starting from $k = 4$ apart from quasi equidistant sets it is possible to create reflected factorial codes $\Phi_{rc}(k)$. Starting from $k = 4$ apart from quasi equidistant sets it is possible to create reflected factorial codes $\Phi_{rc}(k)$. The algorithm of reflected codes creation depends on their number of digits. In particular, here is easily provable by direct verification.

Statement 3. *The set of uniform reflected factorial codes defined by recurrence relation*

$$\Phi_{rc}(k) = 0\Phi_1(k-1) \parallel 1\Phi_1^R(k-1),$$

Let's discuss the problem of synthesis of quasi equidistant factorial codes with a number of digits $n = \overline{4, 7}$. So taking the data from Tab. 3 let's construct a preliminary weights distribution of n – bite code combinations resulting in Tab. 4. The amount of codes with even and odd weights in current table for all variants n are satisfying inequality (3) and this means, that all required conditions for quasi equidistant factorial codes creation are met.

Let's go to validation to the whole amount of trees variants $\Phi_1(5)$. First of all pay attention (Fig. 2) the code combinations with weight of 4 must reside between codes with weights equal 3. This is required to provide a distance between related combinations equal to 1. Merge code pairs with weights equal to 3 among whose code with weights equal to 4 are reside. By that we can get rid of two code pairs with weights 3 and 4 in column $n = 5$ Tab. 4 and schema (8) rewrite as

$$01202020202020202020 \quad (9)$$

There are group of nine odd (O) code combinations which contains four codes with weight equal to 1 and five with weight equal to 3 in the schema (9). It is evident the 126 variant of not complete trees of sequence $\Phi_1(5)$ exists, equal to number of nine by four combinations. And now take into consideration that in each of 126 variants of symbolic form (9) because of the operation, inversed to "merge" operation described above, it is possible to restore entire schemas of trees (8). Because of 10 possible methods of inverse operation means the entire amount of trees $\Phi_1(5)$ construction equal to 1260. Performing by the same method validation of amount of trees $L_\Phi(6)$ of $\Phi_1(6)$ sequences we get $L_\Phi(6) = 1513512$. With increasing of number of digits n the complexity of combinatorial validation $L_\Phi(n)$ and amount of trees $\Phi_1(n)$ dramatically increases. For example, all 10 variants of trees $\Phi_1(4)$ are shown in Tab. 5.

Table 5. Trees $\Phi_1(4)$

№	Tree variant	№	Tree variant
1	012323212121	6	012123212321
2	012321232121	7	012123212123
3	012321212321	8	012121232321
4	012321212123	9	012121232123
5	012123232121	10	012121212323

First of all we construct ranged by weights ν sequence of uniform codes $\Phi(4)$ (Tab. 6).

Table 6. Ranged $\Phi(4)$ codes

№	Code weight ν			
	0	1	2	3
1	0000	0001	0011	0111
2		0010	0101	1011
3		0100	0110	
4		1000	1001	
5			1010	

In correspondence with a schema of sixth tree variant (Tab. 5) the first two code sequences, which will be called *layers* of tree branch, choose 0000 and 0001 codes.

We could choose 0010 layer instead of 0001. The third layer to choose would be a code with weight equal to 2, the one which consist of 0001 code with Hamming distance equal to 1. Suitable ones are codes in columns with 1, 2 and 4 numbers of Tab. 6. The code with smaller number will be considered as a base, the rest – alternative. Keep moving the same way with codes choosing for $\Phi_1(4)$ sequence, using the schema of chosen tree, we have a Tab. 7.

Table 7. Synthesis of of $\Phi_1(4)$ branch

№	Code weight	Base code	Alternative code	
1	0	0000		
2	1	0001	0010	
3	2	0011	0101	1001
4	1	0010		
5	2	0110		

The ninth layer of tree under synthesis should be a code with weight equal to 2, moreover it must reside from previous code with distance equal to 1.

But there is no such a code, which were not used in Tab. 6. In order to cope with this deadlock we will do the following. We will go up through columns of and will do a substitution in this row with a nearest alternative code located from the right of it. In this case we should substitute base code 0011 with alternative code 0101 and afterwards continue the synthesis procedure for $\Phi_1(4)$. An example of quasi equidistant codes $\Phi_1(4)$ synthesized by method of direct enumeration is shown in Tab. 8.

Table 8. $\Phi_1(4)$ Sequences, correspondent to 012321212321 tree

Number of tiers	Tree	The branch of the tree									
		1	2	3	4	5	6	7	8	9	10
0	0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
1	1	0001	0001	0010	0010	0010	0010	0100	0100	0100	0100
2	2	1001	1001	0011	0011	1010	1010	0101	0101	1100	1100
3	3	1011	1101	1011	1011	1011	1011	1101	1101	1101	1101
4	2	0011	0101	1010	1010	0011	0011	1100	1100	0101	0101
5	1	0010	0100	1000	1000	0001	0001	1000	1000	0001	0001
Number of tiers	Tree	The branch of the tree									
		1	2	3	4	5	6	7	8	9	10
6	2	1010	1100	1001	1100	0101	1001	1001	1010	0011	1001
7	1	1000	1000	0001	0100	0100	1000	0001	0010	0010	1000
8	2	1100	1010	0101	0101	1100	1100	0011	0011	1010	1010
9	3	1101	1011	1101	1101	1101	1101	1011	1011	1011	1011
10	2	0101	0011	1100	1001	1001	0101	1010	1001	1001	0011
11	1	0100	0010	0100	0001	1000	0100	0010	0001	1000	0010

4 Fibonacci Sequences

Fibonacci codes are generalized concept of classical binary code [9]. Any nonnegative integer $N = 0, 1, 2, \dots$ can be exclusively represented by a numerical Fibonacci function

$$N = \alpha_n F_n + \alpha_{n-1} F_{n-1} + \dots + \alpha_k F_k + \dots + \alpha_2 F_2 + \alpha_1 F_1 \quad (10)$$

Besides the sequence $\{\alpha_k\}$ in (1) doesn't contain pairs of neighbor unities which are provided by equivalent conversion called "folding" operation: $011 \rightarrow 100$. This operation makes it possible to represent Fibonacci number as so called "minimal" form, the code combination of which will have minimal weight.

For example, [10],

$$\underline{0111} \underline{1011} 001 \rightarrow 100 \underline{1110} 0001 \rightarrow 10100100001 \quad (11)$$

The codes which are underlined in example (11) are codes for which folding operation was performed. As it follows from this example the folding operations resulted in weights decreasing of code combinations. Namely, the amount of units in the final code is less than in the original one.

Using the folding operation it is easy to come to a representational algorithm of multidigit binary Fibonacci numbers. As an example let's consider a method of representation of natural sequence of decimal numbers (including zero) by four digits numbers of Fibonacci codes. We need to agree to label code numbers from right to left assuming the smaller (the very right) number the correspond to number 1, then number 2 and so on. We choose such a coding method of first three decimal numbers 0, 1 and 2:

$$\begin{aligned} 0_{10} &\rightarrow 0000 ; \\ 1_{10} &\rightarrow 0001 ; \\ 2_{10} &\rightarrow 0010. \end{aligned} \quad (12)$$

A conversion from decimal number k_{10} to $(k+1)_{10}$ number in Fibonacci codes (label them as F_k and F_{k+1} correspondingly) will be performed using a rule: if there is 0 in a smaller position F_k then it is substituted with 1 in F_{k+1} code. If there is 1 in a smaller position F_k then this 1 goes to the second position and writes as 0 in a smaller position. This rule is using in system (12) while conversion from F_1 to F_2 .

Let's represent number 3_{10} with Fibonacci code. But before we go, following the rule described above we will get code $3_{10} \rightarrow 00011$ which by folding operation would be represented in its minimal form

$$3_{10} \rightarrow 0100. \quad (13)$$

According to statements (12) and (13), the smaller positions of Fibonacci codes are using for decimal numbers 1, 2 and 3 representations correspondingly. Those values are generalized by the following recurrent block synthesis algorithm of binary Fibonacci sequences. Let's $F(k)$ – is a set of Fibonacci numbers of the same length including 0. Then we have:

Statement 4. *A set of k – bite Fibonacci numbers of the same length is defined by recurrent correlation*

$$F(k) = 10 \parallel F(k-2). \quad (14)$$

The proving of just formulated statement can be easily performed by a method of direct verification. In the right part of (14) the $F(k-2)$ set is consisted of $(k-2)$ – position numbers.

From this it is followed that if any subset of Fibonacci numbers, included in $F(k-2)$, contain digits the number of digits of whose are less than $k-2$ then those numbers are prefixed with required amount of zeros. Algorithm (14) is right for any value $k \geq 2$. Indeed, if $k = 2$ then

$$F(2) = 10 \parallel F(0).$$

As long as $F(0)$ set is empty then $F(2)$ set contains the only Fibonacci digit 10, which corresponds to decimal digit 2_{10} .

There are Fibonacci codes for limited sequence of decimal numbers calculated using recurrent formula considering initial condition (12) in Tab. 9. Zeros, which are located to the left of bigger unit in Fibonacci coders, have been removed.

You can see values n in column F of Tab. 9, equal to number of codes which can be created by a fixed number of binary positions. For example, $F = 3$ means the four bite combinations, which contain 1 in its older position, can be created three Fibonacci codes. Writing down the values from F column we will get sequence 1, 1, 2, 3, 5, 8, 13, ... which is classical sequence of Fibonacci numbers.

Now go to estimation of variants of quasi equidistant Fibonacci code trees. For this purpose based on data from Tab. 9 let's create a preliminary table of distribution of code combinations weights, included in $F(k)$, $k = \overline{4, 7}$, (Tab. 10). By analysis of data from Tab. 10 we have the following conclusion. Quasi equidistant sequences of four digit Fibonacci numbers are end up with code combinations with weight of 1, five or six number of digits with weight of 2 and seven numbers of digits with odd weight equal to 1 or 3.

Table 9. Fibonacci numbers

k_{10}	F_k	F	k_{10}	F_k	F	k_{10}	F_k	F
0	0		13	100000		21	1000000	
1	1	1	14	100001		22	1000001	
2	10	1	15	100010		23	1000010	
3	100	2	16	100100	8	24	1000100	13
4	101		17	100101		25	1000101	
5	1000	3	18	101000		26	1001000	
6	1001		19	101001		27	1001001	
7	1010		20	101010		28	1001010	
8	10000	5				29	1010000	
9	10001					30	1010001	
10	10010					31	1010010	
11	10100				32	1010100		
12	10101				33	1010101		

Table 10. Distribution of code combinations weights $F(k)$

All code combinations	Number of code digits (k)			
	4	5	6	7
0	1	1	1	1
1	4	5	6	7
2	3	6	10	15
3		1	4	10
4				1
n_q	4	7	11	17
n_h	4	6	10	17
All together	8	13	21	34

It is not that complicated to perform a calculation $L_F(k)$ of quantity of variants for quasi equidistant Fibonacci sequence $F_1(k)$ trees.

The result of this calculation for chosen k parameters is shown in Tab. 11.

Table 11. Power of tree subset $F_1(k)$

k	Amount of tree variants $F_1(k)$			
	4	5	6	7
$L_F(k)$	1	5	126	205920

For reflected Fibonacci codes it is right the following

Statement 5. A set of even k bite reflected Fibonacci codes is defined by recurrent correlation

$$\Phi_{or}(k) = 00F_1(k-2) + 10F_1^R(k-2), \tag{15}$$

where $F_1^R(k)$ –sequence is inversed to $F_1(k)$, i.e. the sequence of quasi equidistant codes $F_1(k)$ written in reverse order.

As an example (Tab. 12) of calculated using a computer a branch of one tree $F_1(6)$.

Table 12. Sequences $F_1(k)$ of tree 012321232123232121212

Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
0	0	000000	000000	000000	000000	000000	000000	000000	000000
1	1	000001	000001	000010	000100	000100	001000	001000	010000
2	2	000101	010001	100010	000101	010100	001010	101000	010001
3	3	010101	010101	101010	010101	010101	101010	101010	010101
4	2	010100	000101	001010	010001	000101	101000	100010	010100
5	1	000100	000100	001000	000001	000001	100000	100000	000100
6	2	100100	100100	101000	100001	100001	100001	100001	000101
7	3	100101	100101	101001	100101	100101	101001	101001	100101
8	2	100001	100001	001001	100100	000000	001001	001001	100100
9	1	100000	100000	000001	100000	100000	000001	000001	100000
10	2	100010	100010	010001	100010	100010	010001	010001	100010
11	3	101010	101010	010101	101010	101010	010101	010101	101010
12	2	101000	101000	000101	101000	101000	000101	000101	101000
13	3	101001	101001	100101	101001	101001	100101	100101	101001
14	2	001001	001001	100001	001001	001001	100100	100100	100001
15	1	001000	001000	100000	001000	001000	000100	000100	000001
16	2	001010	001010	100100	001010	001010	010100	010100	001001
17	1	000010	000010	000100	000010	000010	010000	010000	001000
Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
18	2	010010	010010	010100	010010	010010	010010	010010	001010
19	1	010000	010000	010000	010000	010000	000010	000010	000010
20	2	010001	010100	010010	010100	010001	100010	001010	010100

5 Binomial Sequences

There are many known methods for binomial codes creation and based on them – binomial sequences [11]. We will consider two ways of even binomial codes synthesis in this unit. First of them we will call an “algorithm A. Borysenko”, and the second one an “algorithm of A. Beletsky”, which is called as *alternative* algorithm here in after.

The whole idea of first algorithm of uneven binary binomial codes, which correlate to algorithm of full summarized binomial arithmetic, is described in [12], page 124. Of course any uneven binary code can be converted to even code of n number of digits (length). For this purpose it is just enough to prefix the code combination such amount of zeros so the common number of digits became equal to n .

To construct algorithms of binomial arithmetic by Borysenko it is enough to define two parameters k and n , the first one defines the maximal amount of units in codes, the second one by value $r = n - 1$, defines the maximal length of uneven binomial number. A decimal zero in Borysenko’s binomial code is written down as $l = n - k$ of zeros, the range P of binomial numbers is defined by formula $F_{\max} = P - 1$. Here are a number of examples of binomial numbers B_x (algorithm A. Borysenko), creation whose correspond to decimal value x (Tab. 13).

Table 13. Variants of binomial number sequences

$n = 6, k = 4$		$n = 6, k = 2$				$n = 6, k = 3$					
x	B_x	x	B_x	x	B_x	x	B_x	x	B_x		
0	00	10	11010	0	0000	10	10000	0	000	10	1000
1	010	11	11011	1	00010	11	10001	1	0010	11	10010
2	0110	12	11100	2	00011	12	1001	2	00110	12	10011
3	01110	13	11101	3	00100	13	101	3	00111	13	10100
4	01111	14	1111	4	00101	14	11	4	0100	14	10101
5	100			5	0011			5	01010	15	1011
6	1010			6	01000			6	01011	16	11000
7	10110			7	01001			7	01100	17	11001
8	10111			8	0101			8	01101	18	1101
9	1100			9	011			9	0111	19	111

Let’s label $B(n, k)$ – sequence of binomial numbers created by Borysenko’s algorithm. From analysis of Tab. 4 we get the following conclusion.

Statement 6. *Direct and inverse binomial sequences are linked with correlation*

$$B(n, k) \equiv \overline{B}^R(n, n - k),$$

where $\overline{B}^R(n, n-k)$ – sequence of binomial codes, which first of all is written in reverse order to codes in $B(n, k)$ and secondly each position of $\overline{B}^R(n, n-k)$ forms by result of inversion (i.e. substitution of 0 to 1 and vice versa) of corresponding positions $B(n, k)$.

Let’s find out a possibility of quasi equidistant codes $B_1(n, k)$ creation based on set of binomial numbers $B(n, k)$. For this purpose using the data from Tab. 13 lets create a table of code combination weights distribution (Tab. 14) included in $B(n, k)$ set. According to data from Tab. 14 and also values n_e and n_o comparison, received for many other parameters n and k , we can conclude the inequality (3) for codes $B(n, k)$ is not true and as sequence it is true

Table 14. Distribution of code combination weights $B(n, k)$

Weight of code combination	$B(6, 4)$	$B(6, 2)$	$B(6, 3)$
0	1	1	1
1	2	4	3
2	3	10	6
3	4		10
4	5		
n_q	9	11	7
n_h	6	4	13
All together	15	15	20

Statement 7. Binomial codes do not form quasi equidistant sequences.

Let’s move to creation of alternative binomial codes. Introduce numeric function

$$B = \alpha_n C_n^{\alpha_n} + \alpha_{n-1} C_{n-1}^{\alpha_{n-1}} + \dots + \alpha_k C_k^{\alpha_k} + \dots + \alpha_1 C_1^{\alpha_1} \tag{15}$$

where

$$C_l^k = \binom{k}{l} = \frac{k \cdot (k-1) \cdot \dots \cdot (k+1-l)}{l!},$$

- binomial coefficient which is equal to number of k and l combinations. The coefficients α_k are defined by a correlation $\alpha_k = 0, \lceil k/2 \rceil$, in which $\lceil x \rceil$ means rounding of number x to the nearest integer above.

Series (15) is presented in form of binary coefficients α_k for each of who’s the limited number of positions equal to number of digits and required for binary value $\lceil k/2 \rceil$ representation is assigned.

Coefficient unambiguously defines the value of monomial $\alpha_k C_k^{\alpha_k}$, as it is shown in Tab. 15 (in which for example purpose the value $k = 7$ is chosen).

Table 15. An example of monomial series calculation (16)

α_7	0	1	2	3	4
$C_7^{\alpha_7}$	1	7	21	35	35
$\alpha_7 C_7^{\alpha_7}$	0	7	42	105	140

For a sequence of binomial codes created by numerical function (15), let's introduce a label $B(n,r)$ in which n parameter will be called a *power* of a function, and r – *order* of function, which is equal to coefficient α_n . A fragment of binomial codes is shown in Tab. 16.

Table 16. The sequence of binomial numbers $B(4,2)$

N	α_3	α_2	α_1	N	α_4	α_3	α_2	α_1
0			0	10		1	0	1
1			1	11		1	1	0
				12		1	1	0
2		1	0	13		1	1	0
3		1	1					
				14	1	0	0	0
4	1	0	1	15	1	0	0	0
5	1	1	0	16	1	0	0	1
6	1	1	1	17	1	0	0	1
				18	1	0	0	1
7	1	0	0	19	1	0	1	0
N	α_3	α_2	α_1	N	α_4	α_3	α_2	α_1
8	1	0	1	20	1	0	1	0
9	1	0	1	21	1	0	1	0

In order to decide a question regarding the possibility of quasi equidistant binomial sequences creation let's create a table of a set of code combinations weights (Tab. 17).

Table 17. Distribution of weights of code combinations $B_1(n, r)$

Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10
0	1	1	1	1	1	1	1	1
Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10
1	2	2	2	2	2	2	2	2
2	3	5	5	6	6	6	6	6
3	1	2	4	9	9	12	12	12
Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10

A feature of alternative binomial codes is that they do not allow creating quasi equidistant codes in a full manner as it is visible from Tab. 18. In particular, for all sequences shown in Tab. 18, the latest codes (highlighted) reside from previous codes with a Hamming distance equal 3 but not 1, as it is required for sequence $B_1(4,2)$. This feature of alternative binomial codes is visible in all possible variants $B_1(n,r)$.

6 Conclusions

The main result of this research is formation of generalized conditions for quasi equidistant and reflected codes existence which are produced by even consistent binary code combinations in a mixed numeration systems. Except of Gray codes the Fibonacci, factorial and binomial codes with Hamming distance between related code combinations equal to 1, are also included in a set of such codes. The main method for synthesis of quasi equidistant codes is a method of computer direct enumeration. The results of this research can be easily generalized and applied for cases where Hamming distance is more than 1.

References

1. Shannon, C. T.: A Mathematical Theory of Communication. Bell. Syst. Tech. J., 27, 379 – 423, 623 – 656 (1948)
2. Efimenko, V. V., Karpjuk, B. V., Stukalin, Iu. A.: An Algorithm for Synthesis of Binary Quasi Equidistant Codes. Journal of Acad. Science, USSR, AVTOMETRIJA, 5, 109–115 (1968) (In Russian)
3. Bogdanov, G. T., Zinovjev, V. A. Todorov, T. J.: On the Construction of Quasi Equidistant Codes. Journal of Problems of Information Transmission, 43(4), 13–36 (2007) (In Russian)
4. Beletsky, A. Y., Beletsky E. A.: Quasi Equidistant Codes. NAU Publishing, Kiev (2008) (In Russian)
5. Banja, E. N., Selivanov, V. L.: About the Features of the Construction of Various Number Systems. Journal of NTUU "KPI" Informatics, Management and Computer Science, 49, 68–73 (2008) (In Russian)
6. Borysenko, A. A., Cherednychenko, V. B.: Number Systems in Computing. Bulletin of the SSU, Engineering Series, 4, 162–177 (2009) (In Russian)
7. Grey, F.: Pulse Code Communication, Pat. USA, № 2632058 (1953)
8. Borysenko, A. A.: Discrete Mathematic. Textbook publishing house SSU (2007) (In Russian)
9. Stahov, A. P.: Codes of Golden Proportion. Radio Communication, Moscow (1984) (In Russian)
10. Stahov, A., P.: Fibonacci Codes, http://goldenmuseum.com/1010FibCodes_rus.html
11. Zanten, A. Ja.: Binomial System and Enumerations of Combinatorial Objects. Journal of Discrete Analysis and Operations Research, Series 1. 6, 12–18 (1999) (In Russian)
12. Borysenko, A. A.: Binomial Count. Theory and practice. Publishing house SSU (2004) (In Russian)

Mechanism Design for Foreign Producers of Unique Homogeneity Product

Vitaliy Kobets¹

¹ Kherson State University, 1, 40 rokiv Zhovtnya Street, 73000, Kherson, Ukraine

vkobets@kse.org.ua

Abstract. Paper concerns to impact on custom receipts of duty rate changing from single to differentiated ones by customs house for foreign producers. To get maximal custom receipts for achieving of social goal state may introduce differentiated duty rates for foreign producers of unique product. Success of this state policy will depend on effectiveness of incentive compatibility conditions for these producers.

Keywords. Mechanism design, single duty, differentiated duty, custom policy, social choice function

Key terms. MechanismDesign, RevelationPrinciple, SocialGoal, MathematicalModel, IncentiveCompatible

1 Introduction

Mechanism is a mathematical structure, modeling an institute and determining the set of rules, regulating actions accessible to the participants and determining as participants strategies in given communication system are transformed in results. In the absence of co-operation mechanism between participants the final result can substantially differ from social optimal one. A mechanism implements given objective function, realizing it on participants types space [2; 8].

Mechanism structure includes [7]:

1. Social choice function (SCF is a final result demanded by the society)
2. Implementation mechanism (realization of SCF by the payoffs and distributive functions of product and money);
3. Revelation mechanism of participants types (by a social planner);
4. Motivating mechanism (it is intended to make conditions for revelation of true information by participants about their types θ [6]).

Objective function F is a composition of messages μ and result h (fig.1).

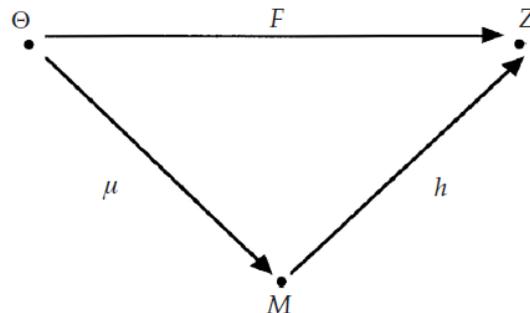


Fig. 3. Mechanism M and objective function F

Mechanism can enforce to cooperation rules, when participants accomplish actions, violating set rules [4]. Two extremes of mechanisms' types are centralized (planned system) and decentralized (such as competition market), between them are continuum numbers of other mechanisms.

The decentralized mechanism (saving confidentiality mechanism) implies the private expenditure (for collection and verification of information reliability) [1]. Existent mechanisms can be complemented or substituted by the new ones, for example, by the means of legislation changes.

Reasons of new mechanism introduction:

- Revelation of unsatisfactory aspects activity of existent economic systems or institutes (market failures)
- Established economic system gives advantage only for certain participants

Mechanism tasks:

- To ground social choice function with the desirable characteristics for society
- To develop compatible conditions for participants to reveal their true types (reservation price, costs etc.)
- To make implementation process of social choice function by the help of chosen mechanism (direct or indirect)

The *direct mechanisms* provide direct transfer of the truthful private information about their types by the agents to the public planner (not realistic mechanism). The *indirect mechanisms* create motive, under which to the agents more profitable to open true information, than to conceal or to distort it (more realistic mechanism) [10].

During organizing of customs mechanism, as well as any other, to the participants concerns of social planner (government) and agents (payers of the customs tax). The agent is selfish person, who has private information only about him or her own type (for example, personal income, costs, profit).

Economic environment is exogenous variable, given by nature or received from the last periods (competition type, technology, rule of custom policy). In model neither the agents, nor mechanism designer do not know prevailing environment. Mechanism designer knows: (i) class of environments, for which should be developed the mechanism; (ii) desired criterions for SCF [5; 9].

SCF represents criteria for result estimation, but not a means of goal achievements as mechanism does.

For customs house SCF mapping types space (average costs of production for importers) in results space (custom receipts). The participant type (average costs) defines its message (invoice cost of the goods), which causes final result (custom receipts).

So the purpose of customs mechanism can be maximization of receipts from customs taxes in the state budget under creation of the appropriate motivating system for the importers (increase of the invoice price, preferential duty rates regime).

This paper has a following structure. We make a literature review in this, first, part. Problem statement and basic assumptions of model are presented in the second part. Part 3 deals with main results for participants under fixed and differentiated duty rates. Last part concludes.

2 Problem Statement

Search of effective ways of state budget replenishment by the means of indirect taxes requires introduction of flexible duty customs for foreign producers foreseen by the proper government laws in relation to payment of custom payments. Criteria, after which the state aims to set the duty rate on import commodities, and to foresee protectionism principle for domestic producers, profitableness principle for the state and utility principle, for domestic consumers.

Peculiarity of optimization for import duty rate is that foreign producers, forming a competition domestic market, will maximize own profits, taking into account a market price [3], whereas for state size of custom rate depends on invoice cost of commodity, which can be corrected by a custom house in the direction of increase and have to corresponds to prevailing (equilibrium) market price.

Product invoice price indicates cost of commodity, which transfers through custom border of Ukraine. If the invoice price indicated in freight customs declaration below of average price in the base of Government custom service of Ukraine, there is the rise of product price to average level before getting customs clearance for product. From the customs value of product duty and VAT is counted, that countries are transferred in a budget.

Customs takes place as follows:

$$B = t \cdot TR_N,$$

where B – duty sum; TR_N – part of product invoice price, that exceeds an un-taxable size (in UAH); t – duty rate (in per cent) from the product invoice price, which now in Ukraine is equal 10%.

For construction of model, that describes co-operation of foreign producers and customs we assume:

- n foreign firms produce homogeneous product, which is supplied to the domestic market and has no domestic analogues;
- between firms there is quantitative Cournot competition;
- cost functions of all firms are linear on production quantities (constant scale return), and reverse domestic market demand function is linear on the quantity of foreign products;
- information about average costs of foreign firms and domestic market demand is uniformly (symmetrically) distributed between *all* participants (foreign producers, domestic consumers and government).

Participants' objective functions:

Foreign producers:

Total cost of producer i consists of variable cost (fixed cost we assume zero in long-run period, v_i is average (variable) cost of producer i) and duty sum:

$TC_i^F = v_i \cdot q_i + t \cdot P \cdot q_i$, where $P_f = P$ is invoice price of unit product.

Profit of producer i : $\pi_i^F = P \cdot q_i - (v_i \cdot q_i + t \cdot P \cdot q_i)$ or $\pi_i^F = (1-t) \cdot P \cdot q_i - v_i \cdot q_i \xrightarrow{q_i \geq 0} \max$, $i = 1, \dots, n$, t is endogenous variable, i.e. duty rate determined by government.

1. State (Ukrainian Income and Duty Ministry)

Tax proceeds to state budget is: $B = t \cdot P_f \cdot \sum_{i=1}^n q_i$.

2. Domestic market:

Reverse linear function of domestic demand is $P = b - c \cdot Q = b - c \cdot \sum_{i=1}^n q_i$,

where P is market price of product, b - maximal price of foreign product on domestic market (under zero import supply).

3 Results

3.1 Custom Receipts Model Construction for Fixed Duty Rate

3.1.1 Producer profit maximization

After substitution of market price to profit function of producer i we obtain:

$\pi_i^F = (1-t) \cdot \left(b - c \cdot \sum_{i=1}^n q_i \right) \cdot q_i - v_i \cdot q_i$, $i = 1, \dots, n$. First order condition (FOC)

First order condition for maximization of custom receipts is determined by condition $\frac{dB}{dt} = 0$ or equivalent to following equation: $bt^2 - 2b \cdot t + b - \bar{v} = 0$, from here equilibrium duty rate will be equal:

$$t = 1 - \sqrt{\frac{\bar{v}}{b}} \tag{3}$$

Equilibrium single duty rate (3) will have inverse relation with average cost of all foreign producers and direct relation with maximal domestic product price.

Thus the invoice price of product will be set at a level $P = b - c \cdot Q$ or taking into account (2) and (3) we will get the equilibrium indexes of invoice price and sales accordingly:

$$P_f^* = \frac{\sqrt{b} \cdot (\sqrt{b} - n \cdot \sqrt{\bar{v}})}{(n+1)}, Q^* = \frac{n\sqrt{b} \cdot (\sqrt{b} - \sqrt{\bar{v}})}{c \cdot (n+1)}$$

Farther from expression $B = t \cdot P_f \cdot Q$ we will define that the equilibrium (maximal) custom sum will form:

$$B^* = \frac{n\sqrt{b} \cdot (\sqrt{b} - \sqrt{\bar{v}})^2 \cdot (\sqrt{b} - n\sqrt{\bar{v}})}{c \cdot (n+1)^2} \tag{4}$$

Consider dependence between equilibrium duty state and custom receipts on fig.2.

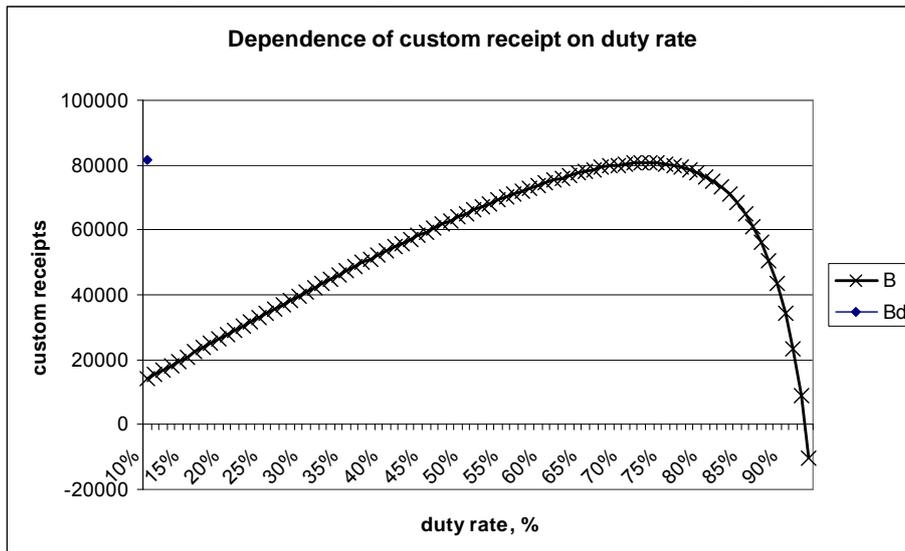


Fig. 2. Laffer curve – dependence between custom receipts and duty rate

$$(n=10, b=40, v=5,25, c=0,01, P=24,04) t=73\%.$$

3.2 Custom Receipts Model Construction for Differentiated Duty Rate

3.2.1 Producer profit maximization

Profit of foreign producer I is presented by next expression: $\pi_i^F = (1-t_i) \cdot P \cdot q_i - v_i \cdot q_i \xrightarrow{q_i \geq 0} \max$, where t_i is differentiated duty rate for foreign producer i. FOC for profit function gives ($i = 1, \dots, n$):

$$\frac{\partial \pi_i^B}{\partial q_i} = (1-t_i) \cdot \left(b - 2c \cdot q_i - c \cdot \sum_{j \neq i} q_j \right) - v_i = 0.$$

Similarly we obtain partial derivatives for profit functions of others foreign producers.

$$\begin{cases} 2 \cdot q_1 + q_2 + \dots + q_n = \frac{b}{c} - \frac{v_1}{c \cdot (1-t_1)}, \\ q_1 + 2 \cdot q_2 + \dots + q_n = \frac{b}{c} - \frac{v_2}{c \cdot (1-t_2)}, \\ \dots, \\ q_1 + q_2 + \dots + 2 \cdot q_n = \frac{b}{c} - \frac{v_n}{c \cdot (1-t_n)}. \end{cases}$$

System solving by matrix approach give optimal sales values for foreign producers on domestic market (duty rate $0 < t_i < 1, i = 1, \dots, n$):

$$q_j = \frac{1}{(n+1) \cdot c} \cdot \left[b - \frac{n \cdot v_j}{1-t_j} + \sum_{i \neq j} \frac{v_i}{1-t_i} \right], j = 1, \dots, n. \tag{5}$$

3.2.2 Budget custom receipts maximization

Receipts from taxation of differentiated duty rates for foreign producers of homogeneity products will equal:

$$B_d = P_f \cdot \sum_{i=1}^n t_i \cdot q_i \xrightarrow{t_i \geq 0, i=1, \dots, n} \max, \text{ where } P_f = \text{const} - \text{invoice price per}$$

unit product for foreign producers.

FOC for maximization of custom receipts is defined by following n conditions:

$$\frac{dB_d}{dt_i} = 0, \text{ where } i = 1, \dots, n. \text{ Obtained n-equation system with n unknown duty}$$

rates t_i after equivalent algebraic transformations define *reaction curves* (6) $t_i = f_i(t_{-i})$, which demonstrate dependence duty rate of i -th producer t_i and duty rates of all its rivals t_{-i} . In this function duty rates for foreign producers have to change in a same direction. Thus increasing optimal duty rate by one of the producers requires rising of duty rates for all others foreign producers.

$$t_i = 1 - \sqrt{\frac{v_i \cdot \sum_{j \neq i} (1 - t_j)}{b + \sum_{j \neq i} \frac{v_j}{1 - t_j}}}, \quad i = 1, \dots, n. \quad (6)$$

Such adjustment change of duty rates will proceed until the equilibrium size of each duty rates will not be set.

System solving of n equation formed from functions (6) gives the following sizes of equilibrium duty rates for foreign producers:

$$t_i = 1 - \sqrt{\frac{v_i}{b}}, \quad i = 1, \dots, n. \quad (7)$$

Obtained result shows *reverse* dependence between the average cost and size of duty rate for import product: $\frac{dt_i}{dv_i} < 0$ and shows *direct* dependence between maximal price of domestic market and duty rate.

From expression (7) follows that more effective producers (with average cost lower than industry average cost) will be assessed after the higher duty rate, than less effective ones for maximization of custom receipts. Now equilibrium invoice price and quantity sale with using of expression (7) and linear function of domestic demand will be set at the appropriate levels:

$$P_f^{d*} = \frac{\sqrt{b} \cdot \left(\sqrt{b} + \sum_{i=1}^n \sqrt{v_i} \right)}{(n+1)}, \quad Q^{d*} = \frac{\sqrt{b} \cdot \left(n\sqrt{b} - \sum_{i=1}^n \sqrt{v_i} \right)}{c \cdot (n+1)}.$$

It is important circumstance that after differentiation of duty rates for the foreign producers, import of product on domestic market will drop $Q^{d*} < Q^*$, that will result in rising of price for consumers. Additionally, possibility of charging lower duty rates for one producer and higher for another ones will generate corruption actions. To prevention it, necessary objective indexes for differentiation of these rates. Expedience of differentiated rate introduction will arise only after condition of increase of custom receipts $B_d > B$ in comparison with the fixed duty rate (fig. 3).

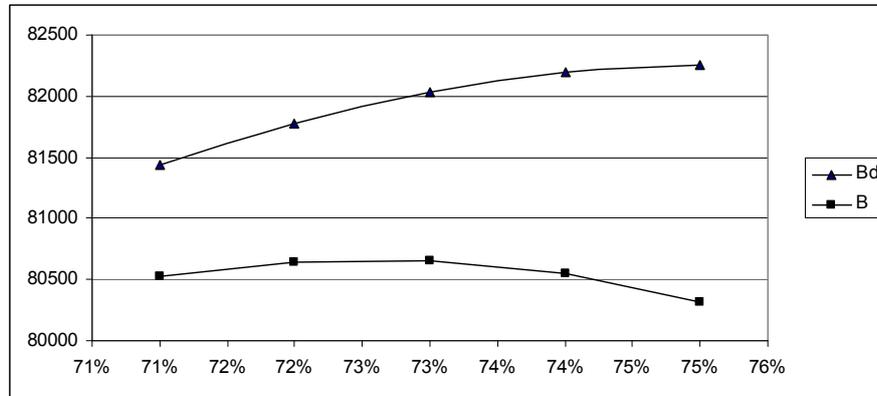


Fig. 3. Comparison of influencing of the differentiated and fixed duty rates on customs receipts ($n=10$, $b=40$, $v=5,25$, $c=0,01$, $P=24,04$)

4 Conclusions

So if as to differentiation of tariffs to take public accountant reports from the financial records audit of firm in part of total cost forming, it will decrease possibilities of realization of unfounded duty rates differentiation by custom house.

At the same time more effective producers will be interested not to disclosure information about true total cost with purpose to drop size of duty rates. Less effective producers vice versa will have motives to reveal its total cost, which below than average cost per unit.

Thus if less effective producers will prove that the effective ones gave false information, it will become foundation for the rise of duty rates to more effective producers and decrease of duty rate for less effective. Thus such custom policy will allow to the state will put information transaction cost about producers from itself on less effective producers. From one's part, more effective producers will have motives to prove that less effective producers set too high the size of its inefficiency.

Collusion between all foreign producers about non-disclosure information about own costs will be highly unlikely when number of foreign producers will be grow and collusion will be high-probability when firm concentration will be high.

To receive maximal custom receipts for achieving of social aim state may implement differentiated duty rates for foreign producers of unique product which depend from producers' cost. Success of this state policy will depend on effectiveness of incentive compatibility conditions for these producers, which mean extracting of true information about cost from foreign producers by the means of firms cross-sectional audit.

References

1. Dilip, M.: Decentralization, Hierarchies, and Incentives: A Mechanism Design Perspective. *Journal of Economic Literature*, 44, 367–390 (2006)
2. Jehle, G. A., Reny, P. J.: *Advanced Microeconomic Theory*. Prentice Hall, New York (2005)
3. Williamson, O. E.: *Markets and Hierarchies: Analysis and Antitrust Implications*. Free Press, New York (1975)
4. Maskin, E.: *Mechanism Design: How to Achieve Social Goals*, HSE, Moscow (2009) (In Russian)
5. Nikolenko, S. I.: *Mechanism Design Theory*, Binom, Moscow (2009) (In Russian)
6. Archibald, G. C.: *Information, Incentives and the Economics of Control*. Cambridge University Press, London (2005)
7. Narahari, Y., Garg, D., Narayanam, R., Prakash H.: *Game Theoretic Problems in Network Economics and Mechanism Design Solutions*. Springer Series in Advanced Information and Knowledge Processing (AIKP). Springer-Verlag London Limited, London (2009)
8. Hurwicz, L., Stanley, R.: *Designing Economic Mechanisms*. Cambridge University Press, Cambridge (2006)
9. Izmalkov, S., Sonin, K., Yudkevich, M.: *Mechanism Design Theory*. *Questions of Economics*, 1, 4–26 (2008) (In Russian)
10. Bergemann, D., Stephen, M.: Robust Mechanism Design. *Econometrica*, 73, 1771–1813 (2005)
11. Myerson, R.: *Game Theory Analysis of Conflict*. Harvard University Press, Cambridge (1997)

Features of National Welfare Innovative Potential Parametric Indication Information-Analytical Tools System in the Globalization Trends' Context

Elena Lazareva

Southern Federal University, 105, str. Bolshaya Sadovaya, 344006, Rostov-on-Don, Russia

e1_lazareva@mail.ru

Abstract. In the article innovation-reproductive and rent-generating function of national welfare is exposed, necessity and real ways of national welfare innovative potential look-ahead analytical estimates methodology and tools' revision in a context of globalization trends are offered, complex analysis of the results of system parametric indication of the strategy of national welfare development in the innovation economic growth interests on the author's set of instruments ground is conducted.

Keywords. National welfare, innovation as a new form of combining industrial, intellectual and social resources; innovation rent; information-analytical tools system

Key terms. NationalWelfare, CorporateModel, SpatialStrategy

1 Introduction

In a context of the modern economic development model the essence of national welfare is expressed in new aspects – it becomes not only the accumulated re-iterative reproduction process result, but also is converted into the integrated innovation-oriented economic growth resource-factor. This conversion is connected with world and national economic systems movement towards innovative «knowledge economy», competition gravity center transference to the science, education, innovative activity sphere, non-material actives role in economic reproduction process increasing.

The resource-provided countries have the export-raw model of economy. Their development may be characterized in comparison with other countries by the rough, spasmodic rate, mainly caused by considerable raw materials prices and economic instability. Such development is inevitably accompanied by the problems which brake economic modernization and its social and innovative orientation. On the contrary, the development of the countries which realize the policy of human capital quality,

national well-being, high technologies increment provides advantages in world socio-economic evolution, raises competitiveness of national «intellectual» economy.

Increasing human development quality importance for economic growth generating and competitiveness initiated the mounting interest of economists to the subjective factor (the human capital) role in production progress. It gradually promoted the national welfare parameters (at first – the individual, especially economic; later – the social, public) inclusion into the economic dynamics resource supply research system.

The globalization accompanied by substantial capital mobility and national economies openness increasing transforms the national welfare economic content and display forms in reproduction process, modernizes its structure and functions in the conditions of transition to the innovative-focused economy.

These tendencies find reflection in the new long-term economic trend research methodology – the methodology which equally considers society and economy interests. The national welfare becomes the major productive forces element and the integrated institutional condition of the human capital reproduction.

The world financial-economic crisis, showing critical dependence of the national economies upon mobile global resources (financial, information, technologic resources) and, in particular, exposing the fact that the dynamics of the Russian GDP is still to a considerable extent determined by the external factors of the conjuncture, made topical the problem of finding internal, innovative resources-sources for development. The present situation requires the internal innovative resources of social-economic development, first of all, existing resources of the national welfare, the reserves of which in Russia are still unused in full due to the underdeveloped nature of the institutions of their conversion into competitive factors of production, active usage.

In this light, especial topicality is attributed to the issues related to technical-methodological analysis of the national welfare resources in the system of the global competitive resources, to determination of their role in the process of social innovative reproduction, conditions and mechanisms of their conversion into the innovative factors of production as well as integrated evaluation of the human capital of the country, its efficient usage and higher rate of innovation oriented development of the economic subjects and of economy overall.

The strategy of the economic subjects policies' economic-oriented modernization has to be based, due to the aforementioned facts, upon evolutionary-cyclical, informational-innovative paradigm of the economic development theory and upon resource analysis, in accordance with which national welfare in the postindustrial society plays the role of an integrated resource for the innovative economic trends. One witnesses not only a different nature of the input of national welfare into the reproduction process, but its various composition, i.e. apart from traditional material elements, which have cost measurement (revenue level, volume and structure of the personal consumption fund etc.), greater importance is attributed to its social elements – level and quality of education of the population, level of its health, housing conditions, degree of security within the society, quality of the social-ecologic habitat, social capital, social-economic mentality, condition of general and spiritual culture in the society, set of the symbolic benefits etc., which do not have market cost and, often, which have the nature of social benefits, i.e. they create general social conditions for fulfillment of a person, for creative freedom.

In accordance with the above, the economic subjects' innovative social-economic policies have to comprise not only the innovative processes direct support strategies and mechanisms, but also person-oriented, comfortable general social conditions for innovation-oriented development of economy creation, realized in the form of welfare, better life standards and insuring efficient reproduction of the human capital [1].

Orientation towards bigger human and social capital and, consequently, bigger investments in the anthropo-social capital in the process of intellectual resources social reproduction as integral parts of the national welfare, constitute the basis for forming its innovative, resource-reproducing functions. It follows that the principal problem of continued innovative economic development consists of the national welfare into the innovation-initiating human and social resources – factors of production inherent to the “economy of knowledge” social-economic conversion mechanisms. Such the national welfare components institutional conversion into the innovative resources, human and social benefits economic composition is their comparative advantages (competitiveness) capitalization within the framework of countries' integration in the world economic relations, i.e. transformation of said advantages into the source of the added value and objects of the global companies, business, integrated structures, states innovative activities.

The innovative rent, which is received due to national welfare reproduction and its conversion into the innovative-intellectual production economic resources, constitutes an economic basis of the innovation oriented development. Within the framework of the cluster theory, “network economy” – the innovative economic rent plays the role of the result of the national welfare components, situated in the country, efficient usage. The variety of the innovative rent categories is due to different categories of benefits – resources of national welfare, which constitute the source of the rent formation.

The research is based upon such founders of the innovation-oriented economic development theory as D. Bell, A. Buzgalin, V. Inozemtsev, N. Kondratjev, S. Kuznets, B. Kuzyk, G. Mensch, B. Milner, R. Nizhegorodtsev, D. North, V. Ovchinnikov, J. Osipov, D. Tis, E. Toffler, J. Schumpeter, J. Jakovets etc.

The national welfare potential evaluating scientific basis of the welfare economy in the innovation-oriented development of economy system is considered in the works of the following authors: J. Bentham, S. Valentej, L. Walras, A. Marshall, L. Nesterov, V. Pareto, A. Pigou, A. Smith, J. Hicks, L. Erhard etc., who analyze the problems of the benefits value, of wealth formation, its distribution, conditions for market balance as a principal factor for social welfare, problems of harmonizing individual and social welfare judging by different criteria. D. Buchanan, J. Galbraith, J. Mill, W. Eucken, J. Rawls, V. Cherkovets, R. Ehrenberg etc. analyze a great number of social-economic factors, which affect the social welfare growth in the market economy.

Different aspects of the national welfare role identifying in the of innovation-oriented economic development system are researched in the works of P. Aguilon, R. Barro, A. Varshavskij, J. Vinslav, S. Glazjev, I. Diskin, J. Coleman, V. Kostjuk, D. Lvov, V. Makarov, N. Moisejev, N. Rimashevskaja, D. Rodrik, S. Rosefelde, K. Salomon, A. Sen, R. Solow, J. Stiglitz, M. Todaro, F. Fukuyama etc. Study of their works allowed specifying scientific interpretations of national welfare from the new institutional evolutional cyclical paradigm of economic development point of view.

The nature and specificity of national welfare systems functioning taking into account their correlations with innovative development of economy and its different institutional structures were researched by A. Auzan, P. Drucker, V. Ivanter, G. Kleiner, A. Prokhorovskij, V. Tambovtsev, F. Hayek, J. Jasin etc.

The methods of national welfare resources parametric evaluation are studied in the works of S. Ajvazjan, G. Becker, N. Zubarevich, I. Maslova, M. Mozhina, R. Nurejev, L. Ovcharova, V. Polterovich, J. Rjumina, A. Shevjakov etc. Applicable mechanisms and decision making technologies in the sphere of national welfare resource management are analyzed in the works of M. Baskova, O. Bogomolov, A. Dynkin, M. Musin, O. Pchelintsev, S. Rosenfeld, S. Sampler, V. Tretjak, T. Schultz, M. Jagolnitsjer, etc.

Theoretical analysis of such phenomena as “informational civilization” (R. Abdejev, S. Djatlov, M. Kastels, S. Parinov, F. Jansen), “national innovative systems” (K. Bagrinovskij, M. Bendikov, O. Golincheko, I. Dezhina, J. Lotosh), “intellectual capital” (E. Brooking, A. Gaponenko, M. Malone, T. Sakaya, L. Edwinsson), “cluster development strategy” (T. Anderson, A. Weber, M. Iversen, A. Isaksen, N. Kaljuzhnov, R. Kachalov, J. Christensen, B.-A. Lunvall, A. Ljamzin, L. Markov, N. Nagrudnaja, P. Nertog, L. Nesta, M. Porter, M. Enright) was also important for paper’s conception making.

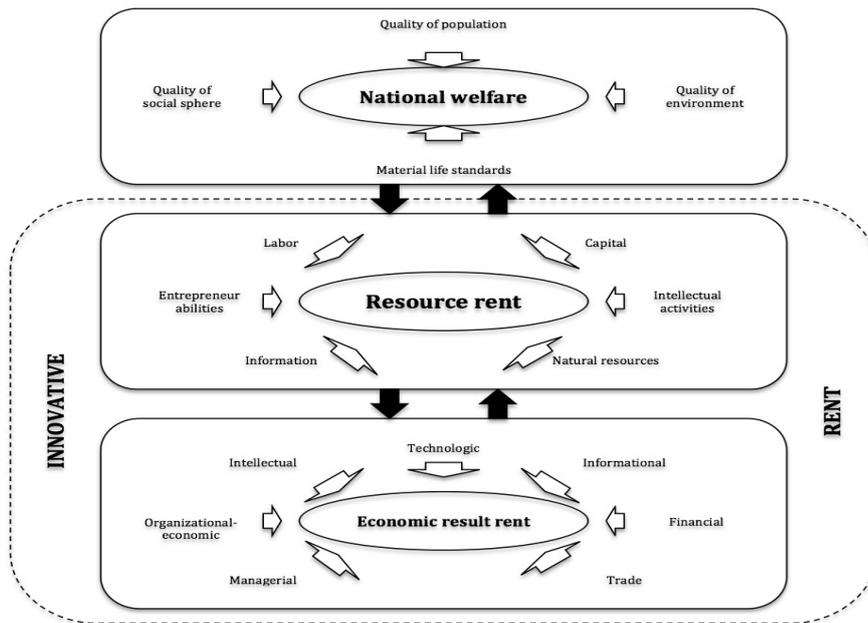


Fig. 1. National welfare as the innovative process rent-yielding factor

Acknowledging high importance of the aforementioned scientists research and noting the fact that there are fundamental approaches for exposing separate facets of the topic considered in this article, it is necessary to underline, however, that hitherto one has not realized an approach related to a complex evaluation of the national welfare as an integrated resource of the nation oriented economic development, one has not ex-

posed its innovative-reproductive function within the framework of involving into the economic system such factors as knowledge and intellect of the nation. Little practical research has been made with regard to the issue of modernizing mechanisms of its conversion into innovative economic resources.

Insufficient conceptual-methodological development of the resource approach to the national welfare analysis in the system of the innovation-oriented economic development; resource support of the innovation vector of economic development in conjunction with its theoretical-applicative topicality determined the purpose of the research.

2 Intermediate Results of Stages of Research

The purpose of the article is to form a methodological basis and to elaborate a theoretical-conceptual model and information-analytical tools system of the national welfare innovation-reproduction function, conditions, mechanisms and implements of using its resources in the interest of developing an innovative-economic system complex analysis. Achieving the set goal determined the necessity and logical sequence of solving a set of stage-by-stage theoretical-applicative tasks. The results of fulfilling said tasks could be formulated as follows:

1. Innovation-oriented development of the present day national economies within the framework of long-term global evolutionary trends is more and more determined by the national welfare level and its dynamics. The national welfare resources accumulation induces higher volume and quality of human capital, higher labor efficiency, modernization and efficient innovation-oriented national economies development.

2. The national welfare structure encompasses not only traditional material benefits-resources characteristic for pre-industrial and industrial societies (real monetary revenue, volume and structure of the personal consumption fund, housing conditions, employment etc.), moreover, it includes new benefits-resources, having higher marginal utility (level/quality of education and health of the population, quality of the social-ecologic habitat, freedom of access to new technologies and scientific discoveries, social capital etc.). The definitive result of mentioned factors involving into the innovative productive cycle is revenue in the form of innovative rent creation which insures competitiveness of the entire production process.

The economic composition of such national welfare resources conversion is capitalization of their competitive advantages in the course of countries integration in the world market and network world-economic relations, especially, in the high-tech spheres, based upon high quality of the human capital, in other words, transformation of said advantages into the source of the added value and into the objects of global investment activities. Within the framework of the innovation-oriented dynamics, national welfare thus assumes the function of its resource-factor, increment of which within the world and state structure of social-economic relations becomes a key prerequisite for the innovative economic development trend.

3. Globalization processes exert contradictory influence upon economic mechanisms related to the national welfare resources usage aimed at the support of innovation-oriented development. On one hand, they broaden the innovative-economic space of the country and the possibilities of converting the resources of its national welfare

competitive potential into innovative-intellectual resources, on the other, globalization brings about an additional impetus for bigger inter-state asymmetry, polarization of the countries innovative development. Macroeconomic indicators analysis characterize the level of the countries integration into the global innovation-oriented economy, which showed a high level of interstate developmental inequity and lack of competitiveness of a set of components of the national welfare. This situation decreases the level of converting separate components of the national welfare into innovative economic development resources.

4. The national welfare resources accumulation-consumption (reserve-flow) values correlation is a distinctive indicator of the innovative reproduction process cyclical development. During stagnation, low rate of economic dynamics, accumulated national welfare is depleted (as the result of mobilizing its certain part in order to insure innovative economic growth), and during rise, high growth rate, the situation is opposite, national welfare is accumulated due to added national revenue, creating thus an integrated basis for a long-term incremental trend of social economic innovation-oriented development.

5. Greater role and larger scale of the economic development innovative factors change traditional perception of the classic stages of the modern expanded production. The stage of accumulating intangible assets - factors of production, which create the innovative economy resource basis becomes the initial and principal stage in the new scheme of reproduction economic relations. This stimulates national welfare accumulation with the view of achieving higher productivity, first of all, of the intellectual resources, of human capital, creation of the institutional habitat, beneficial for elaboration and distribution of innovations, and due to these factors higher rate of innovative economic dynamics.

6. The need to convert of the accumulated tangible and intangible national welfare resources into innovative development factors is embodied in the new priorities and strategies of the long-term state economic policy, in accordance with which the innovative growth of economy is due to observance of the principal of correlation and balance of the imperatives of economic efficiency, social justice and ecologic stability as the three principal criteria of the innovation-oriented reproductive development of a high aggregation level. Moreover, this includes the fact that the state innovative policy assumes a new objective function – the function of balanced social-economic interests of the national, regional (local) and global economic subjects in the process of accumulation, reproduction and usage of the national welfare resources with the view of innovative growth.

Studies of the new model of subject-object relations in the national welfare reproduction system and conversion of its elements into resource sources for innovative growth showed that the “network reality” conditions makes topical the issue of elaboration of a collective strategy for the the national welfare development, in which the aspect of “co-operation” prevails over the aspect of “competition”, and the classical model of the civil society, based on legal definitions of liberalism and market regulation, is replaced by the corporate community model (fig. 2).

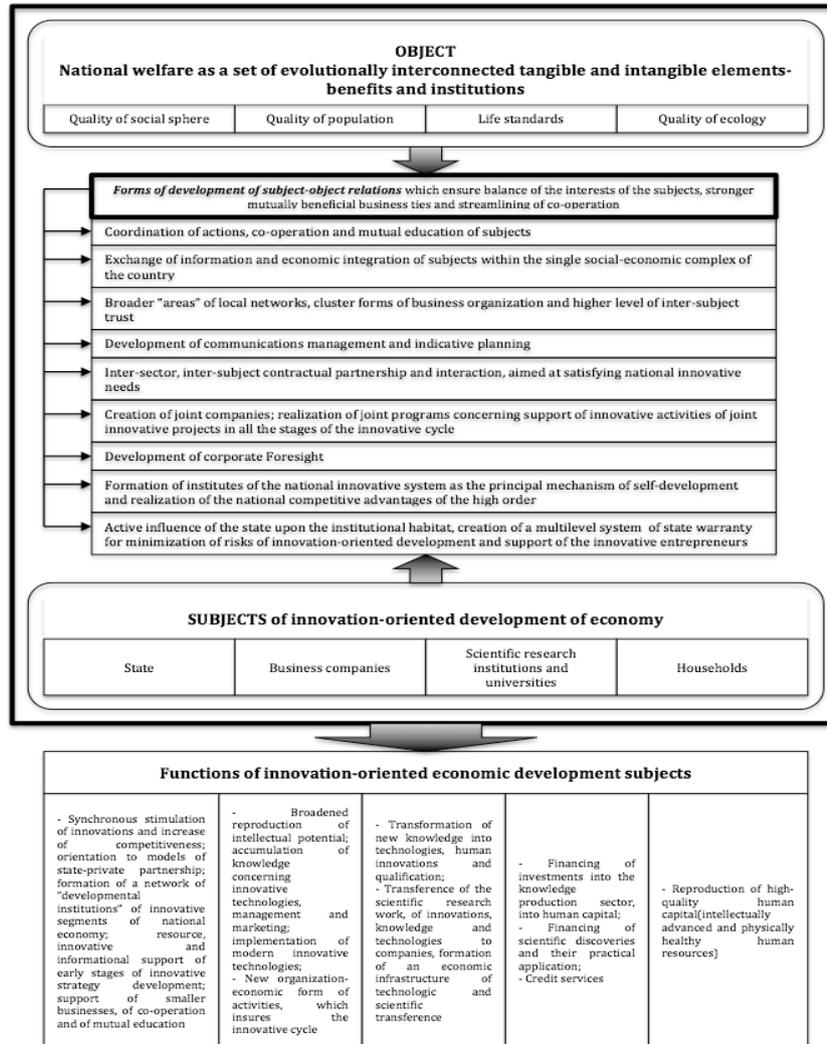


Fig. 2. Subject-object relations' conceptual model

Coordination and balance of the specific interests of innovation-oriented development subjects in reproduction of national welfare as a national benefit, based upon joint advantages, trust and state-personal partnership make one of the important methodological principles for the corporate strategy formation, which helps to optimize management and to achieve higher efficiency of national welfare usage with the view of national economic growth. Determination of the ideal "hierarchical chain" of interests of the economic subjects and orientation of the adequate stimulation policy towards it is one of the alternatives for realizing the coordinative methodological principle.

7. The national welfare resources inclusion into the economic asset balance of the country signifies their interpretation as a source for added value in the long-term innovative cycle of economic dynamics.

The most adequate approach to economic evaluation of tangible and intangible components of national welfare is a modified variant of the Hartwick-Solow principle, in accordance with which it is necessary to consider the innovative rent as the principal source of the national revenue, a part of which is channeled into national welfare accumulation, that brings about a higher resource potential of the long-term innovation-oriented economic development – it is re-invested in better quality of the human capital and amelioration of the social-ecologic conditions of its reproduction (education, healthcare, fundamental science, social infrastructure, lower environment pollution).

8. Rent revenue, which is received due to efficient usage of the new categories of intangible benefits – informational, innovative, infrastructural, intellectual benefits that directly insure reproduction of human capital, play a greater role in the national economies innovation-oriented modernization of national economies. As it is shown by the analysis of said processes in Russia, taking into account the existing institutional deficits, underdeveloped nature of venture business, lack of a systematic, high-quality network habitat, favorable for diffusion of innovations and weak interest of the economic subjects in their elaboration and implementation, capitalization of the present innovative potential of national welfare (infrastructural, educational-intellectual, informational welfare) is rather difficult, whereas innovative rent is gained only in separate, isolated cases. As the result a considerable part of the existing tangible and intangible national welfare resources, first of all, intellectual and human resources, is not capitalized. This fact brings about a lower competitiveness of the country.

9. Need to indicate and insure elaboration of the national welfare components conversion into innovative economic resources mechanisms presupposes analysis and evaluation, to be executed in the state management system, regarding the level and dynamics of reproducing its four components – quality of the population proper, material life standards of the population, quality of the social habitat and quality of ecologic state of the natural-economic complex. Complex evaluation of the aforementioned four components of the national welfare with the view of achieving innovative economic growth is based methodologically upon usage of sophisticated theoretical-analytical set of implements including a set of formalized methods and models of determining latent connections between national welfare and innovation-oriented economic growth (which form a unity of the innovative reproduction process) as well as evaluation of the innovative effects due to a higher level of converting different national welfare components into factors of innovative growth.

The elaborated set of implements allows to analyze efficiency of the existing national welfare resource structure, to expose its limiting components and to form on this basis a strategy for a long-term economic policy, aimed at development of institutions, which increase national welfare resources competitiveness and the level of their conversion into productive sources of innovative economic growth.

A distinctive feature and advantage of the elaborated model set of implements is the possibility to use it in order to accumulate analytical information regarding the results and parameters of economic, social, ecologic strategies related to accumulation

and increment of the national welfare resources with the view of achieving a higher national economic dynamics and to thus provide (as opposed to the traditional implements) a more adequate evaluation of the mechanisms used in the state economic policies related to support of the innovation-oriented economic development trends (figure 3) [2].

10. Diagnostics effected on the basis of the set of implements with regard to the national welfare as an integrated resource of innovation-oriented economic development of Russia state-of-the-art within the global coordinates framework (figure 4) and integral innovative effect of its increment showed that due to realized innovative welfare management strategies (including strategies of a higher level/quality of education and lower sickness rate of the population, higher buying power of its monetary income per person and lower level of poverty, development of the social infrastructure, higher social-territorial mobility and level/conditions of employment of the population, development of smaller business and greater freedom of entrepreneurs, creation of a dynamic information infrastructure and better access to technologies and science etc.), the indicators of the Russian economy subjects may be increased approximately by 1.5 times mostly by means of better social sphere quality [3].

The innovative effect due to national welfare increment and its transformation into innovative economy resources indicators: level of economic subjects' innovative activities, level of conversion of national welfare into innovative growth competitive factors, innovative rent capitalization level are the key parameters which characterize the proportions between accumulation and consumption of the national welfare resources state policy.

The quotients, calculated (with regard to Russia) with the view of proposed structural model empiric verification and reflecting the dependence between the dynamics of the innovative activity of the economic subjects parameters and the parameters of the national welfare resources (average expected lifetime, GDPPPP per person, Gini index and ecologic stability index) showed that, within the integrated effect indicator among the four basic components of national welfare, greater importance is held by the social sphere resources which reflects the priority of social, socially-advantageous benefits – social capital accumulation sources reproduction. These sources are characterized by such important properties from the point of view of the innovative growth as: positive network effects and their higher marginal utility in the course of their use; therefore, the level of the national welfare resources into factor sources of innovative growth conversion greatly depends upon the state of the social sphere – the elasticity quotient was 1.724 and the correlation quotient was 0.671, then, following the order of lower dependence one has the ecologic habitat quality – accordingly 0.463 and 0.324, quality of the population – 0.137 and 0.393 and the material life standards – 0.057 and 0.442.

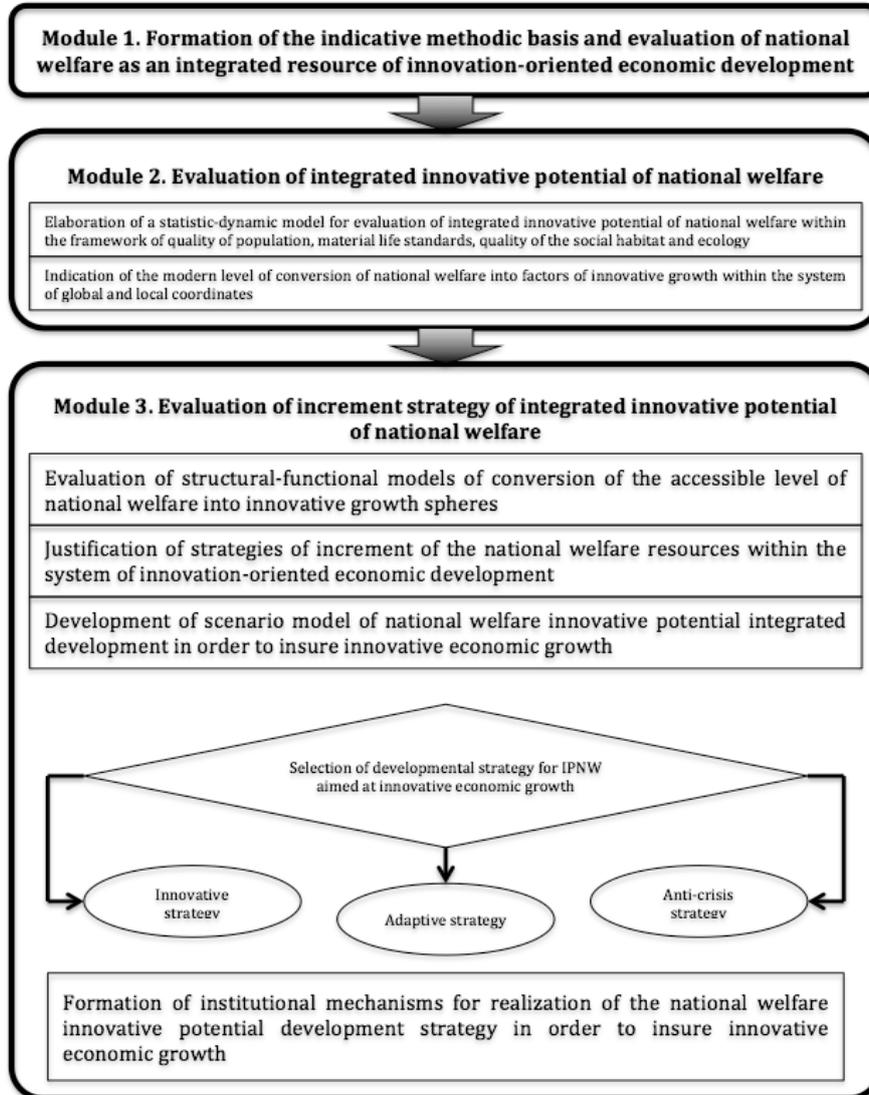


Fig. 3. Model set of implements for state strategy' analytical evaluation

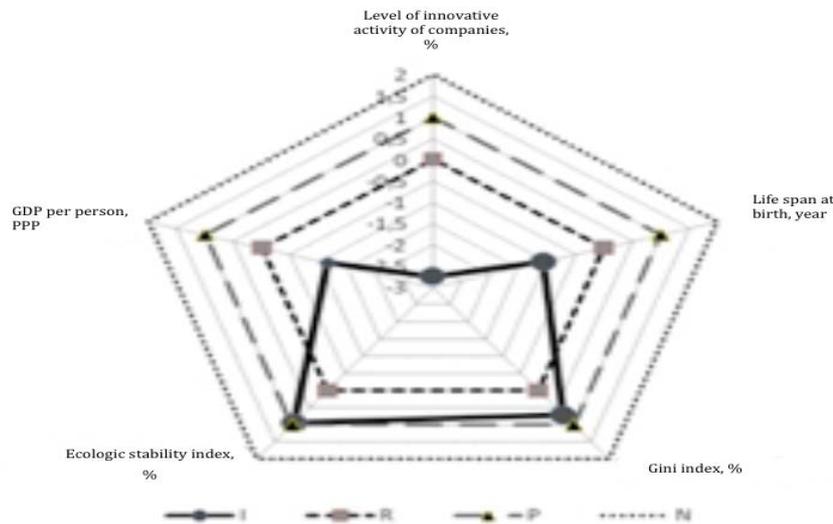


Fig. 4. National welfare state-of-the-art estimation

11. In the course of interregional comparison of the conditions and existing limits for realizing the policy of state national welfare resources reproductive proportions between accumulation and consumption optimization, one detected a domineering correlation between the economic subjects innovative activity indicator and indicators of the achieved level of conversion. The calculations effected by the author according to special methods showed, in particular, the following typology of dependence of the features of the economic subjects innovative activity on the parameters of domineering kinds of national welfare resources (social sphere resources) into innovative growth factors conversion which determine the priorities of the long-term state economic policy: for an economy, which is characterized by a low, medium, high fully realized dependence of the features of the economic subjects innovative activity upon the parameters of the social sphere resources conversion, priority belongs, therefore, to the strategy of developing social infrastructure and higher level/quality of employment of the population, strategies of developing smaller business and greater freedom of entrepreneurs, strategies of easier access to scientific achievements and to new technologies, information infrastructure development (figure 5). The detected innovative effects indicate the priorities of the social-economic policy, in which the main role belongs to investments into the innovative national welfare resources: housing conditions, social and information infrastructure, science, education, healthcare, culture etc.

3 Conclusions

The obtained results show that due to social conditions, factors and motives of behavior more important role, social capital resources greater importance, it is necessary to elaborate a harmonized systematic program of innovation-oriented long-term economic policy modernization and to create a favorable social-economic climate in the country on the basis of the existing national welfare.

The systematic approach means a reconsidered hierarchy of social-economic priorities within the framework of the person oriented innovative economic growth paradigm. In this light the state has the following tasks of paramount importance: amelioration of the overall conditions of employment and population housing, recreation of the salary reproductive function (first of all, on the basis of adequate evaluation of the level/quality of education); accelerated development of the intangible investment complex and social infrastructure, realization of human-saving social programs; consistent industrial policy which would activate innovative activity mechanisms and socially responsible behavior of the corporate subjects which are capable of making their contribution into development of national welfare and human potential of the nation.

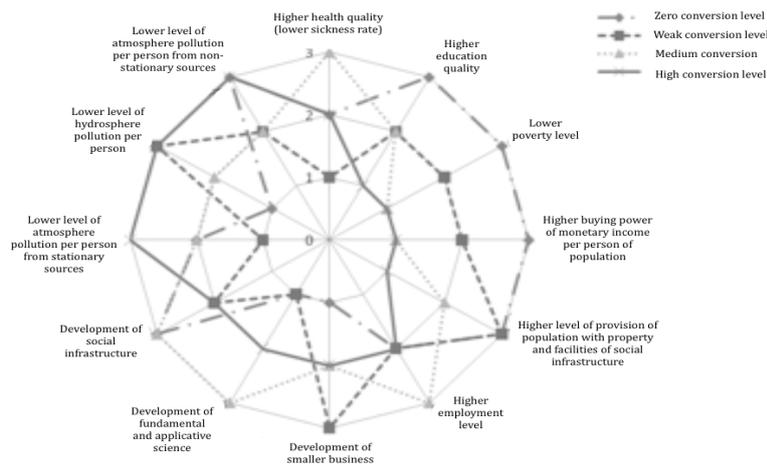


Fig. 5. Spatial strategic “developmental crystal” of Russian national welfare

In the present situation efficient mechanisms of balanced innovation-oriented economic development may be formed only on the basis of the state, civil society and business integrated efforts in order to achieve consistent expansion and rectification of opportunities for the representatives of different social, professional and territorial population groups via reproduction of the national welfare resources as a social benefit. This has to be reflected in the system of innovation-oriented long-term social economic development strategic management.

The national welfare resources will be realized in an efficient way within the framework of an innovative economy, only if there is a stable need for them from the reproductive process. The strategic task is to bring about long-term correlation of the national welfare resources demand and supply in the innovative development of the economy.

The developed theoretical analytical tool allows to evaluate not only efficiency of accumulation and usage of the national welfare resources, moreover, it makes possible to determine the innovative effect due to a higher level of their conversion into the sources of innovative growth.

References

1. Lazareva, E. I.: National Welfare as an Integrated Resource of Innovation-Oriented Development of Economy. Theoretical-Methodological Aspect. Publishing House of the Southern Federal University, Rostov-on-Don (2009) (In Russian)
2. Lazareva, E. I.: Strategy of National Welfare in Interest of Innovative Economic Growth Development: Results of Systematic Parametrical Indication. Economic Newssheet of the Rostov State University, 3, 65–74 (2010) (In Russian)
3. Aleshin, V. A., Lazareva, E. I.: National Welfare Increment as the Imperative Institutional Determinant of Regional Systems' Development in the Innovative Processes' Globalization Context. Social Inequality and Economic Growth, 4, 9–18 (2012)

Matrix Analogues of the Diffie-Hellman Protocol

Alexsander Beletsky¹, Anatoly Beletsky¹ and Roman Kandyba¹

¹Department of Electronics National Aviation University of Kiev,
1, av. Cosmonaut Komarov, 03680, Kiev, Ukraine

alexander.beletsky@gmail.com, abelnau@ukr.net,
romankandyba@mail.ru

Abstract. This paper presents a comparative analysis of several matrix analogs of the Diffie-Hellman algorithm, namely, Yerosh-Skuratov and Megrelishvili protocols, as well as alternative protocols based on irreducible polynomials and primitive Galois or Fibonacci matrices. Binary matrix is primitive, if the sequence of its powers in the ring of residues mod 2 forms a sequence of maximum length (m – sequence). Offer alternative protocols and discuss ways to improve the reliability of their.

Keywords. Encryption key exchange protocol, the irreducible polynomials, a primitive element of Galois field, primitive binary matrix

Key terms. Research, CryptographyTheory, MathematicalModelling

1 Introduction

The Diffie-Hellman algorithm (DH-algorithm) [1] assumes that two subscribers – Alice and Bob both know the public keys p and q , where p is a large prime number, and q is a primitive root. Subscriber Alice generates a random big number a , computes $A = q^a \bmod p$ and sends it to Bob. In turn, Bob generates a random big number b , computes $B = q^b \bmod p$ and sends it to Alice. Then subscriber Alice raises number B received from Bob to her random power a and calculates $K_a = B^a \bmod p = q^{ba} \bmod p$. Subscriber Bob acts similarly, calculating $K_b = A^b \bmod p = q^{ab} \bmod p$. It is obvious that both parties receive the same number K because $K_a \equiv K_b$. Then Alice and Bob can use this number K as a secret key, e.g. for symmetric encryption because a foe who intercepts numbers A and B faces with virtually unsolvable (in a reasonable time) the problem of calculation K , under the condition, that numbers p , a and b were chosen big enough.

2 Yerosh-Skuratov Protocol

In order to form a secret encryption key in the public network by subscribers Alice and Bob, the authors [2] propose to use DH protocol in the cyclic group of matrices $\langle M \rangle$, and the matrix M is considered as public information. It is assumed that Alice generates a random index x , calculates the matrix M^x and sends it to Bob. In turn, Bob generates a random index y , calculates the matrix M^y and sends it to Alice. Then both subscribers raise the matrices obtained from a partner in their secret powers and calculate the sheared matrix (encryption key) $K = M^{xy} \equiv M^{yx}$. The matrix M must be a high-order matrix (at least 100); so, the authors assert (by the way, without a proof), cracking key has invincible complexity. However, in [3] it has been proved, that Yerosh-Skuratov protocol can easily be cracked based on the generalized Chinese remainder theorem.

3 Megrelishvili Protocol

The essence of the protocol [4] is following. Binary initialization vector V and primitive matrix M of order n are accepted as a public key. Subscriber Alice generates a random index x , calculates the vector $V_a = V \cdot M^x$ and sends it to Bob. In turn, Bob generates a random index y , calculates the vector $V_b = V \cdot M^y$ and sends it to Alice. Then Alice computes the key $K_a = V_b \cdot M^x = V \cdot M^{y+x}$, and Bob computes the key $K_b = V_a \cdot M^y = V \cdot M^{x+y}$. It is quite obvious that using such data exchange protocol, both parties receive the same private key K , because $K_a \equiv K_b = K$.

The algorithm of generating the matrices in Megrelishvili protocol is fairly simple and can be explained by the following calculation scheme

$$M_1 = 1, \quad M_3 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & M_1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad M_5 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & & & & 0 \\ 0 & M_3 & & & 1 \\ 1 & & & & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \dots \tag{1}$$

As it follows from (1), the matrices $M_i, i = 1, 2, \dots$, are matrices of odd order only that can cause some difficulties when they are used in cryptography. This shortcoming was remediated by replacing matrices of type (1) by primitive matrices of an arbitrary order that is synthesized based on the so-called generalized Gray transforms [5]. The essence of these transforms is explained below.

The matrix form of direct (for simplicity denoted by number 2) and inverse (denoted by number 3) classical Gray transforms (codes) [6] can be presented in the form

$$2 := \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad 3 := \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad (2)$$

where as an example, the order of the matrix n is set $n = 4$.

Matrices (2), which we call left-sided Gray transform matrices, are in correspondence with the right-sided transform matrix defined by the following relations:

$$4 := 121 = 2^T; \quad 5 := 131 = 3^T, \quad (3)$$

where

$$1 := \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad (4)$$

is the matrix (operator) of the inverse permutation.

The set of operators (2) – (4), supplemented by the operator 0, or e (identity matrix), forms a complete set of simple Gray operators. From the elements of simple Gray operators, one can form so-called composed Gray codes (CGC) generated by the product of simple (elementary) Gray codes. The simplest examples of CGC 121 and 131 can be seen in (3). Both simple and composed Gray codes have a number of remarkable properties. Firstly, the corresponding transformation matrices are nondegenerate and, therefore, are reversible. Secondly, there are simple inverting algorithms for CGC. And, finally, there are “crypto-order” CGC which have the property of primitiveness. Examples of such codes are given in Tab. 1.

Table 1. Gray Composite codes delivering binary matrices property of primitiveness

The order of the matrix (n)			
32	64	128	256
2244424	22533435	2425535	22533435
2442224	22534335	2433534	22534335
12242253	24334225	2435334	24334225
12242443	25224334	22524224	25224334
12252242	222524424	22533334	222535224

Suppose M is a primitive binary matrix generated by the CGC G . With respect to such matrices, the following assertion can be easily proved by the test method.

Assertion. *The primitiveness of matrices M is invariant to the group of linear transformations Ω of the CGC G generating matrix M and transformations of similarity Π over these matrices.*

The Ω –group includes the following operators: cyclical shift, assess statement, inversion and conjugation as well as arbitrary combinations of these operators. Transformation II forms matrix M_p , which is similar to M and determined by the relation

$$M_p = P \cdot M \cdot P^{-1},$$

where P is a permutation matrix.

4 Alternative Protocols

This section proposes two options for alternative matrix protocols of secret key exchange on the open channel of communications. The procedure for the formation of the encryption key K in the first version of the protocol is based on the use of two public and one private key for both subscribers. As a public key a binary initialization vector V of n order and any irreducible polynomial (IP) φ_n of n order are chosen.

Private keys are primitive (forming) elements ω of the Galois field $GF(2^n)$ over the IP φ_n , from which the subscribers (Alisa and Bob) form the primitive secret transformation matrices $G_{\varphi_n}^{(\omega_a)}$ and $G_{\varphi_n}^{(\omega_b)}$ respectively. The element ω of the field $GF(2^n)$ is primitive over IP φ_n , if the minimum rate e , at which $(\omega^e \equiv 1) \bmod \varphi$ assumes the value $e = 2^n - 1$.

Matrix $G_{\varphi_n}^{(\omega)}$ we call Galois matrices. The synthesis of algorithm for such matrices is explained on a concrete example. Let's IP $\varphi_8 = 100101101$, and the generating element (GE) of subscriber Alisa $\omega_a = 111$. We obtain

$$A = G_a = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}. \tag{5}$$

According to (5), the procedure of filling in the matrix G_a is carried out under the following scheme. First, the GE ω_a is arranged in the bottom row of the matrix. The elements of this row in the left from the GE elements are filled with zeros. Subsequent rows of the matrix (in the direction from bottom to top) are produced by a shift of previous lines. If left element of shifted line is 0, then the cyclical shift by one bit to the left (circular scrolling clockwise). In the case where the left element of shifted line is 1, the conventional shift of the line on one bit to the left and 0 is written to the vacant right element in line. Digit capacity of these lines is one bit more than the order of the matrix. The vectors corresponding to these lines are given to the residue

modulo IP φ_n that returns them the capacity, which coincides with the order of the matrix n . Subscriber Bob forms similarly the Galois matrix $B = G_b$ using his primitive element ω_b .

The introduced Galois matrices have some interesting properties. First, the matrix product is commutative, i.e. $A \cdot B = B \cdot A$. At the same time, secondly, if at least one of the GE is not a primitive of the IP, the commutative property of matrices is lost. Based on the above properties of Galois matrices a key exchange protocol was proposed.

We consider that initialization vector V and the IP φ are known. Alice chooses a secret primitive over φ GE ω_a , forms a Galois matrix A , calculates the vector $V_a = V \cdot A$ and sends it to Bob. In turn, the subscriber Bob selects a primitive GE ω_b , forms a matrix B that calculates the vector $V_b = V \cdot B$ and sends it to Alice. After that, both parties multiply vectors obtained from the partner, in own secret Galois matrix. Thus, a shared secret key K will be formed by the fact that the product of primitive Galois matrices over the same IP φ is commutative, and this implies the identity

$$K_a = V_b \cdot A = V \cdot B \cdot A \equiv K_b = V_a \cdot B = V \cdot A \cdot B.$$

Instead of Galois matrices G , Fibonacci matrices F can be used in the protocol with the same success. Fibonacci matrices are associated with Galois matrices by equation

$$F \xleftrightarrow{\perp} G, \text{ or } F = G^\perp; \quad G = F^\perp,$$

where \perp – means the operator of right transposition, i.e. transposition with respect to the auxiliary diagonal matrix.

In the second alternative embodiment of the protocol the secret key K is computed in two rounds. In the first round, which repeats the above-considered first version of the protocol, a common to both subscribers secret binary vector of n – th order V_p is formed. On the basis of this vector, Alice and Bob compute the common permutation matrix P . One can propose different ways of constructing matrices P . Let us consider one of them. Let's $n = 8$ and N is the decimal equivalent of the vector V_p . The task is to create permutation matrix P_8 of order eight for value N . Choose one or another way of numbering elements of matrices P_8 from 0 to 63. Calculate the value $n_8 = N \bmod 64$ and write 1 in that element of the matrix, whose number is equal n_8 . After that, delete from the matrix P_8 the row and column, which contains 1. We obtain a matrix P_7 of 7-th order, whose elements are numbered from 0 to 48. Find the value $n_7 = N \bmod 49$, which is determined by the location 1 of the matrix P_7 and, consequently, in the matrix P_8 . Following the proposed method, one can simply construct a permutation matrix P of any order.

Let proceed to the second variant of the encryption keys protocol. This protocol uses two public keys, which are the initialization vector V , and the irreducible poly-

nomial ω , and also two private keys. These keys are generated by Alice and Bob as a random primitive over \mathbb{F}_q ω and ν . The protocol runs in two rounds. In the first round based on public keys V , ω and secret ω network operators calculate the total permutation matrix P . The second round is performed in the following order. Alice chooses a primitive over \mathbb{F}_q ω_a , forms Galois matrix A_ν , then similar matrix $A_p = P \cdot A_\nu \cdot P^{-1}$, computes a vector $V_a = V \cdot A_p$, and sends it to Bob. In turn, Bob chooses a primitive over \mathbb{F}_q ω_b , forms Galois matrix B_ν , then similar matrix $B_p = P \cdot B_\nu \cdot P^{-1}$, computes a vector $V_b = V \cdot B_p$ and sends it to Alice. After that, both parties multiply vectors obtained from partners on their secret similar Galois matrix. Thus, the shared key K will be generated due to the fact that the matrices A_p and B_p maintain the properties of primitiveness and commutativity of primary matrices A_ν and B_ν , respectively.

5 Protocol of Vagus Keys

One of the major drawbacks of alternative algorithms key generation algorithms for open key cipher infrastructure, in particular the mentioned above the way of synthesis Galois matrix (by the diagonal fill method), is that it could be easily compromised. To prove that, let's see the vector

$$V_a = V \cdot G_{f_n}^{(\omega_a)}, \tag{6}$$

created by Alice.

By the theory of polynomials of one variable x , we know that product of any polynomial $\omega_n(x)$ power of n by x is equivalently either simple shift of polynomial for one bit left or incrementing the power of polynomial,

$$x \cdot \omega_n(x) \rightarrow \omega_{n+1}(x). \tag{7}$$

Taking formula (7), let's represent the Galois matrix $G_{f_n}^{(\omega_a)}$ the power of n by,

$$G_{f_n}^{(\omega)} \pmod{f_n} = \begin{pmatrix} x^{n-1} \cdot \omega \\ x^{n-2} \cdot \omega \\ \dots \\ x \cdot \omega \\ \omega \end{pmatrix} \pmod{f_n} = \omega \cdot \begin{pmatrix} x^{n-1} \\ x^{n-2} \\ \dots \\ x \\ 1 \end{pmatrix} = \omega \cdot E = \omega, \tag{8}$$

where E – the unit matrix.

From formulas (6) and (8) we can get,

$$V_a = V \cdot \omega_a \pmod{f_n}, \quad (9)$$

where all parts are known, except ω_a . Solving the equation (9), we found:

$$\omega_a = V_a \cdot V^{-1} \pmod{f_n}. \quad (10)$$

For example, let's use the matrix $G_{f_n}^{(\omega_a)}$, given by expression (5), where $n = 8$, $\omega_a = 101101$, $f_8 = 101001101$, so f_8 – is public, ω_a – is private keys of protocol. As initialization vector we choose $V = 11010010$, that corresponds to invert by modulus f_8 vector $V^{-1} = 110010$. By formula (9) we get $V_a = 10111111$. Putting the V_a and V^{-1} is the right side of expression (10) and taking modulus f_8 of vectors multiplication results, enemy (Eva) is getting private key ω_a of Alice. The same way, Eva could found secret key ω_b of Bob. After secret keys ω_a and ω_b are found it's trivial to calculate secret key K .

The security of alternative protocols could be increased up to security level of algorithms based on problem of factorization of modular multiplication of big numbers if we assume that there is secret parameter θ , both known to Bob and Alice.

The modification of protocol [6] is the be following. Assume, there are authorized subscribers that have secret parameter θ as binary vector of n – order. Parameter θ could be transported from Alice to Bon (or otherwise), e.g. by RSA protocol. Alice is generating random of n – order number ω_a and computing generating element

$$\theta_a = \omega_a \cdot \theta \pmod{f_n}, \quad (11)$$

by means of generating element Alice is forming Galois matrix $G_{f_n}^{(\theta_a)}$, calculating vector $V_a = V \cdot G_{f_n}^{(\theta_a)}$ and sends it to Bob. In the same way, Bob send to Alice vector

$$V_b = V \cdot G_{f_n}^{(\theta_b)}, \text{ where } \theta_b = \omega_b \cdot \theta \pmod{f_n}.$$

As it shown above, generating elements θ_a and θ_b could be easily computed, so authorized subscribers Alice and Bob, but not Eva, could calculate secret parameter ω of partner. As example, by formula (11) Bob calculates $\omega_a = \theta_a \cdot \theta^{-1} \pmod{f_n}$, that gives him and Alice ability to calculate secret key $K = \omega_a \cdot \omega_b \pmod{f_n}$. Key K as well as any function of it, could be taken as a secret parameter $\theta^+ = K$ for session key generation for public key cipher channels.

We call that way of key generation – protocol (algorithm) of vagus keys. Vagus keys algorithm could be used in both motioned above protocols. The major benefit of vagus key generation algorithm is protection from "man in a middle" type of attack. It's been archived by including in Galois matrices key generation elements of secret element θ , known only by Bob and Alice. In case of secret element θ is changed

by element θ_e of Eva, makes it impossible to Eva to calculate parameters ω_a, ω_b as well as general cipher key K .

6 Conclusions

The article analyzes the known matrix algorithms for exchanging encryption keys between subscribers of a network of open communication channels. The algorithms are based on the modified asymmetric Diffie-Hellman protocol. The essence of the modification is reduced to replacing the large prime numbers of Diffie-Hellman algorithm by assurance nondegenerate primitive binary matrices of high order. Methods of synthesis of these matrices are proposed based on both the generalized Gray codes, and irreducible polynomials. New key exchange matrix protocols have been developed. The protocols developed are superior for cryptographic strength to known cryptographic protocols, particularly Yerosh-Skuratov and Megrelishvili protocols described in this paper.

The proposed variants of vector-matrix protocols for exchanging by cryptographic keys on open communication channels have a good prospect to be applied for symmetric encryption in computer networks protected from the substitution of data, providing the necessary level of protection of private keys from unauthorized access. These protocols can make a strong competition to more resource-intensive RSA protocol.

References

1. Diffie, W., Hellman, M. E.: New Directions in Cryptography. IEEE Transactions on Information Theory, IT-22(6), 644–654 (1976)
2. Eros, I. L., Skuratov, V. V.: Addressing Message Transmitting Using Matrices Over GF (2). Problems of Information Security. Computer Systems, 1, 72–78 (2004) (In Russian)
3. Rostovtsev, A. G.: On the Matrix Encryption (Criticism Yerosh-Skuratov Cryptosystem), http://www.ssl.stu.neva.ru/psw/crypto/rostovtsev/Erosh_Skuratov.pdf (In Russian)
4. Megrelishvili, R. P., Chelidze, M. A., Besiashvili, G. M.: Unidirectional Matrix Function - High-Speed Diffie – Hellman’s Analog. In: Proc. 7-th Int. Conf. Internet - Education - Science 2010. VNTU, Vinnitsya, 341–344 (2010) (In Russian)
5. Beletsky, A. Ja., Beletsky, A. A., Beletsky, E. A.: Gray Transformations. V.1. Fundamentals of the theory. V. 2. Applied aspects. NAU Publishing House, Kiev (2007) (In Russian)
6. Beletsky, A. Y., Beletsky, A. A.: Synthesis of Primitive Matrices over a Finite Galois Fields and their Applications. Information Technology in Education: Collected Works, 13. Kherson: KSU, 23–43 (2012) (In Russian)

Are Securities Secure: Study of the Influence of the International Debt Securities on the Economic Growth

Darya Bonda¹ and Sergey Mazol²

¹ Belarus State Economic University, Minsk, Belarus

bondadasha@gmail.com

² Academy of Public Administration, Minsk, Belarus

mazols@yandex.ru

Abstract. The paper studies the interdependence of the amount of international debt securities, amounts outstanding by country (borrowers) and the GDP growth by country. The author have chosen 34 countries, that represent every region included in the BIS classification, that is developed countries, offshore centers, developing Europe, Latin America, Asia and the Pacific and Africa. It was found that the excessive amount of such type of securities in comparison with GDP leads to slowdown in the economic growth next year.

Keywords. International debt securities, Economic growth, Financial crisis

Key terms. Development, MathematicalModel

1 Introduction

The last financial turmoil has revealed the drawbacks of the existing global financial system. Surprisingly, the worst crisis since the Great Depression has offered a range of opportunities to the world society: to examine the system, exclude “toxic” elements and introduce new methodology to financial regulation.

During the last decades new avenues for financing were creating, deepening the financial system aside from widening the choice of monetary instruments [1] that have caused overestimation of assets and, consequently, financial collapse.

Despite this issue is under thorough control of Bank of International Settlements, Securities and Exchange Committee, International Derivative and Swap Association, Securities Industry and Financial Market Association, every scientists, analyst, governor, outstanding person and a regular student has its own interpretation of how the crisis works, its causes and consequences.

One of the reasons for the growing financial instability was the excessive amount of various types of securities both in national economies and international arena as well as the complexity of the securities issued. New types of financial instruments usually at first are accepted as great invention of humanity, then, especially during recessions are usually blamed for crisis for speculation reasons [7]. After the recovery, they are still widely spread all over the world. Futures, options and other derivatives have experienced such an attitude [4]. International debt securities are considered to be a financial instrument. The amount of securities outstanding in 2007, i.e. country's liabilities, could prevent countries from sustainable growth in 2008.

To the author's point of view, it is reasonable to study the interdependence of the amount of International debt securities, amounts outstanding by country (borrowers) and the GDP growth by country. The presence of such interdependence can allow us to criticize this type of securities and advise the countries to minimize their usage for the sake of sustainable economic growth.

2 Results

The authors are analyzing the interdependence of the amount of international debt securities outstanding in 2007, and the economic growth, expressed in GDP index in the research.

Debt security is a negotiable financial instrument serving as evidence of debt [5]. The statistics on international debt securities issues cover long-term bonds, notes, short-term money instruments [2]. Debt securities include government bonds, corporate bonds, CDs, municipal bonds, preferred stock and collateralized securities (such as CDOs, CMOs, GNMA's). Debt securities may be protected by collateral or may be unsecured, which underlines the importance of scrutinizing them as one of the key instruments of securitization. Collateralized debt obligations are considered to be a risky instrument as far as their coupons and principal repayments are dependent on a diversified pool of loan and bond instruments, either purchased in the secondary market or from the balance sheet of an original asset owner (Handbook of Securities Statistics). Consequently, through assessing the value of underlying assets, collateralized debt obligation as well as other asset-backed securities spread risk while diversifying it, meanwhile creating a range of credit derivatives. These instruments are widely used to make the debt more liquid and make the money lent work as if they were not borrowed and, furthermore, get a margin. Therefore, the author finds it crucial to pay significant attention to this kind of financial instrument as a mean of spreading risk of insolvency of an entity within the international scale.

The BIS definition of international securities (as opposed to domestic) is based on three major characteristics of the securities: the location of the transaction, the currency of issuance and the residence of the issuer. International issues comprise all foreign currency issues by residents and non-resident in a given country and all domestic currency issues launched in the domestic market by non-residents [2].

GDP in current price, purchasing power parity, is the second element of the research. It was chosen as an indicator of the national output, combining real and finan-

cial sector, thus reflecting the size of the economy. The amount of international debt securities can be compared to the GDP, as both indicators reveal the capacity of the countries' economies.

In the figure 1 one can see historical correlation between the international debt securities and GDP in current prices (data from [1], [2], [3], [8]), which shows that the interdependence between 2 components really exists, in addition, during 2002-2008 the line shows different slope. In 2004 and 2005 the inclination is lower, which means the slowdown in GDP growth rate and IDS amounts and, on the contrary, in 2007 and 2008 the graph indicates the rise of the world economies.

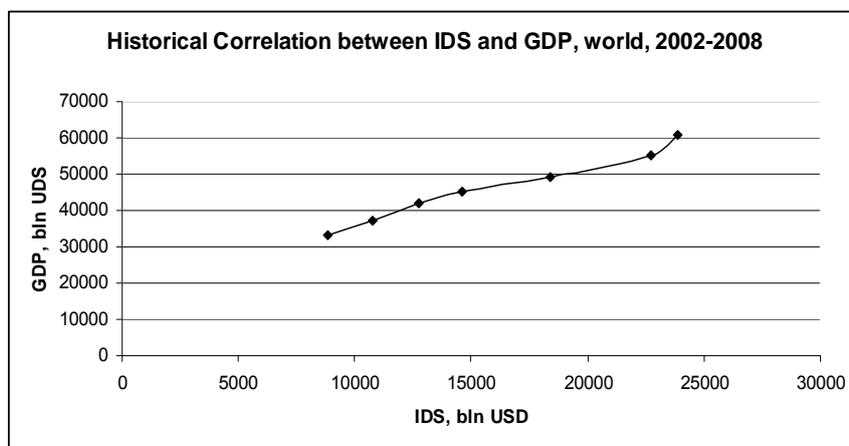


Fig. 1. Historical correlation between IDS and GDP, world 2002-2008

As mentioned in the title, the graph shows the world's tendency. The last economic crisis has damaged major economies leaving some small and developing economies untouched, despite of its scale. This means that by-country analysis is necessary to provide the real evidence of such correlation.

The authors have chosen 34 countries that represent every region included in the BIS classification that is developed countries, offshore centers, developing Europe, Latin America, Asia and the Pacific and Africa. Although the most variation in the amount of securities outstanding had been noticed while scrutinizing the data from developed countries the data utilized represents each continent. Africa is represented by Egypt, Lebanon, Saudi Arabia and UAE, Asia and the Pacific – the Philippines, Singapore, Japan and China, Developing Latin America – Argentina, Brazil, Colombia, Costa Rica, Developing Europe – Belarus, Bulgaria, Czech Republic, Estonia, Russia, Ukraine, offshore centers – by the Bahamas, Developed economies – by Australia, Austria, Belgium, Canada, Finland, France, Germany, Greece, Iceland, Italy, Norway, Spain, Sweden, the UK and US. The X, independent variable, is the amount of international debt securities by country outstanding in 2007 divided by the nominal GDP in billion of USD in 2007, whereas the dependent variable is GDP growth rate in 2008 in comparison with 2007. The linear regression is shown on figure 2.

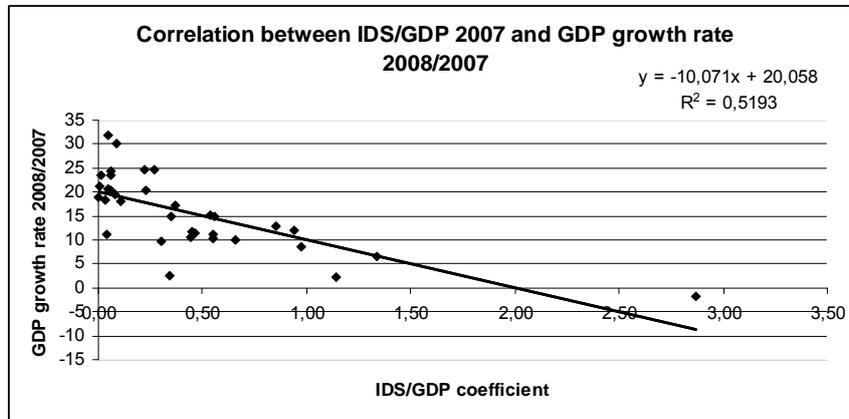


Fig. 2. Correlation between IDS/GDP 2007 and GDP growth rate 2008/2009

The model is described by formula 1:

$$GDP = 20,058 - 10,071 * COEF \tag{1}$$

where GDP – nominal growth rate in 2008 compared to 2007;
 COEF – IDS/GDP coefficient as a ratio of IDS amounts outstanding 2007 and GDP (in current prices, power of purchasing parity, 2007)

The slope is negative, which means the opposite correlation between variables, the bigger the coefficient in 2007, the lower the GDP growth rate in 2008, which makes sense and confirms the author’s theory. The correlation coefficient, which shows the fraction of relationship between variables is -0,72 or 72% of opposite relationship. Therefore, the relation is considered to be strong. The determination coefficient is 0,52 or 52%, which means the variation of the dependent variable is explained by 52% by the independent one.

The linear regression has been chosen because the goal of the author was to find the existence of the relationship between variables, and in addition, negative one, however, admitting the complexity of the relationship, not exactly finding the most appropriate one.

Among the obstacles which prevent linear model from being “best fit” are, first of all, the different level of financial market development and, thus, the vital need for specific financial instruments usage. For example, Belarus is not yet ready for developing credit derivatives, besides, the amount of the securities outstanding has been the same for a couple of years, that means that the amount of securities is not the principal reason for economic crisis. Another reason is overstated prices in some countries which result in the high inflation level, which increases the GDP index far too high to depict the precise relationship between variables. Examples can be Ukraine, Russia and Argentina. As far as offshore centers are concerned, their GDP doesn’t always

illustrate the real production but the value of financial operations, so they need individual approach. However, there are some deviations in the model: for example, Iceland – at the lowest point, because of negative GDP growth, or the US and The UK with the lowest GDP growth in 2008/2007 and highest amount of securities within examined countries.

In the real economic world one can hardly obtain “pareto efficient” outcomes, thus no sector of the economy can be better off without making another worse off [6]. Therefore, the increase of the international debt securities, i.e. rise in liquid liabilities, results in lowering the GDP growth rates. Even though the GDP measures only market production and cannot totally used as a measure of country’s well-being [6], a better measurement of economic activity hasn’t been offered yet. Thus, the influence of financial sector on the overall economy proves the existence of “systemic externality” [6] of the financial market.

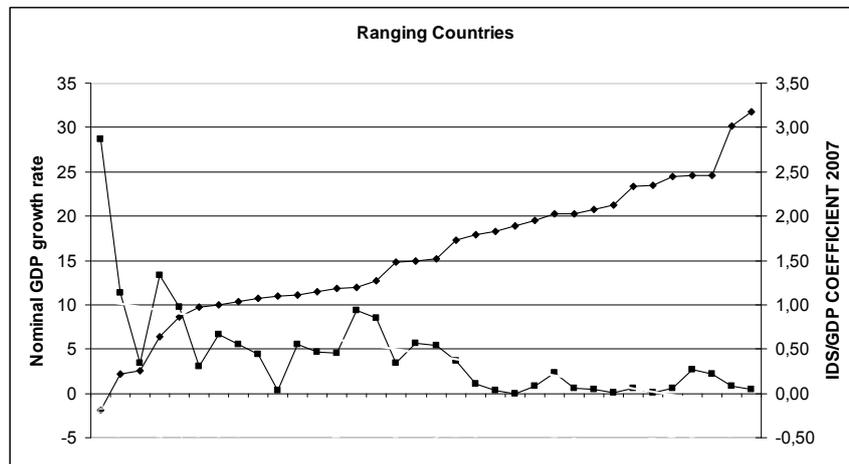


Fig. 3. Ranging countries

The figure 3 is built with the purpose to prove the main idea of the research: the bigger the volume of IDS outstanding (here IDS/GDP coefficient to make it adjustable to different economy sizes), the lower the GDP growth rate. Though, the graph doesn’t totally reflect the inverse dependence, it shows the tendency and proves the idea for the majority of countries: while the slope of the GDP growth rate curve is positive, the slope IDS/GDP coefficient curve is negative.

Among the factors that prevent the ideal illustration of the theory is the nominal character of the variables. One of the elements is the security market capitalization which can not possibly be turned into real one, therefore the usage of nominal GDP growth rate is reasonable. Another factor is the uniqueness of economic development of the countries, like the inflation and unemployment level and the unique interdependence of the economic sectors. Consequently, it is necessary for the countries to develop its own targeting rules, using its own IDS/GDP ratio and work out the model of the development that will be adequate for a single economy specifications.

Perhaps, it is quite possible to work out the limit in IDS/GDP ratio which countries shouldn't exceed in order not to affect sustainable development, but this work requires, first of all, individual approach and demands profound knowledge on economic development of each country.

3 Conclusions

The overall significance of the model is to show the impact of the issuance of international debt securities on the economic growth of the countries. Since the slope of the regression line is negative, the excessive amount of such type of securities in comparison with GDP leads to slowdown in the economic growth next year. Due to the fact that the issuance of securities is partly managed by governments and financial organizations (in case of the US – Securities and Exchange Committee (SEC), for example), the issuance of them can and should be regulated. After the first wave of financial crisis has gone some rating agencies, for example, S&P have offered downgrading system for some risky instruments, thus protecting the market from distributing of excessive risk, as well as SEC has been promoting a new bill to the White House for a while (Reuters). This information shows that global financial society is already trying to react and eliminate some of the causes of the financial crisis, however, not yet fruitful.

The global financial crisis has proved the necessity of the permanent monitoring and control of the financial system, especially with regard to financial architecture and innovations [6]. The research has revealed that the volume of the international debt securities should be a subject of the country's guideline to optimal control and efficient rules for financial policy. Increase in the IDS/GDP ratio may lead to financial instability and increased involvement in the global financial market. In addition, secure financial instrument usage with careful risk control limits allows financial and, therefore, the whole economic system to reach sustainable development.

References

1. Bank for International Settlements: Financial Market Developments and their Implications for Monetary Policy. №39. BIS, Basel (2008)
2. Bank for International Settlements: BIS Guide of Securities Statistics. №39. BIS, Basel (2009)
3. Bank for International Settlements: BIS Quarterly Review, December 2009. BIS, Basel (2009)
4. Chibrikov, V.: History of the Derivatives. Economic Issues, 3 (2008)
5. International Swaps and Derivatives Association: Handbook on Securities Statistics, ISDA (2007)
6. Stiglitz J.: Regulation and Failure in New Perspectives on Regulation, In: The Tobin Project, pp. 11–24, Cambridge (2009)
7. Suetin A.: About the Reasons for Financial Crisis. Economic Issues, 1 (2009)
8. World Bank Database, www.worldbank.org

How to Make High-tech Industry Highly Developed? Effective Model of National R&D Investment Policy

Oksana Moiseeva¹ and Sergey Mazol²

¹ Belarus State Economic University, Minsk, Belarus

oksuta13@mail.ru

² Academy of Public Administration, Minsk, Belarus

mazol.s@yandex.ru

Abstract. The paper validates the relations between the share of public and private R&D spending and the effectiveness of national R&D sector. It states that in order to implement effective and profitable “high-tech policy”, governments have to intensify the share of business sector in Gross Domestic Expenditures on R&D. At the same time it is necessary to preserve the definite “government share” in R&D investments, as reduction of it up to certain extent gives the negative effect.

Keywords. High-tech industry, R&D investment policy, Private v. Public R&D Spending, Exports of High Technology Products

Key terms. Development, Mathematical Model

1 Introduction

It is impossible to deny that people all over the world benefit from new technologies which lead to healthier lives, greater social freedoms, increased knowledge and more productive livelihood. Each day sees additions to the literature, much of which includes reports on the establishment or expansion of R&D facilities and programs that are designed to take the best advantage of highly qualified resources. Nowadays there are practically no governments and politicians that would miss a chance to stress the importance of innovations in economy. According to the judgments of some experts, GDP growth of developed countries up to 50-90% is determined by technological progress and innovations [7]. The developing countries, in their turn, extremely need competitive high-tech industry, not only because being usually one of the most profit-making and cost-efficient industries it contributes to economic prosperity by itself, but also because technological achievements give them a chance to promote and make competitive on the global arena all other economic sectors, narrowing in this way the economic gap between the highly-developed countries and the developing ones.

Still the results of R&D policy in the countries of post soviet space frequently leave much to be desired. The most prominent achievements in the sphere of industrial R&D belong to the most developed countries such as USA, Japan, European Union. Current literature is replete with reports on the expanding R&D activities in China, India, South Korea and Singapore. Meanwhile Belarus, Ukraine, Uzbekistan, Moldova still cannot boast prominent commercial achievements in R&D. That's why this research paper aims at analyzing and investigating of those factors and incentives that turn national innovative efforts, resources and potential into visible and profitable high-tech results.

Research and development (R&D) comprise creative work undertaken on a systematic basis in order to increase the stock of knowledge, including knowledge of man, culture and society, and the use of this stock of knowledge to devise new applications [2].

The UNESCO Institute for Statistics claims that the clearest trend in global R&D activity between 1996 and 2005 was the increasing percentage of GDP devoted by countries all over the world to R&D (R&D intensity has more than doubled in 9% of the countries surveyed, including China, Thailand, Tunisia and others; in 48 out of 89 countries surveyed the percentage of GDP devoted to R&D has significantly increased) [4]. Surely, sustained R&D investment is a key to economic growth. But those are strong words that are easy to follow in good economic times, but more difficult to follow in bad economic times. R&D expenditures are among the first to be cut during recessions. Preliminary data (official statistics on R&D are available only until 2007) suggest that companies have reduced their R&D investment in the aftermath of the crisis. In 2008 the industrial companies despite the challenging economic times continued growing their R&D budgets, expanding by nearly 6,1%, or more than \$60 billion, from what they spent in 2007. Despite their good intentions, when the downturn turned from mild to severe, industrial firms were forced to cut their R&D budgets. Total industrial R&D spending dropped by 1% or nearly \$10 billion overall in 2009 from what was spent in 2008 [6]. These findings are consistent with historical trends showing that R&D expenditure exhibits larger variations than gross domestic product (GDP) over the business cycle. Hence, any drop in GDP would result in an even larger decrease in R&D expenditure [3].

The 2010 Global R&D Forecast, created by Battelle analysts and the editors of R&D Magazine, predicts overall global R&D will increase 4.0% in 2010 to \$1,156.5 billion from \$1,112.5 billion spent in 2009. This increase will mostly be driven by continued spending by China and India, who will drive a 7.5% increase in Asian R&D. American R&D spending is expected to increase 3.2% to \$452.8 billion, while EC spending will only increase 0.5% to \$268.5 billion in 2010. This forecast especially stresses a trend of falling the spending of both the Americas (U.S., Canada, Mexico, Brazil, and Argentina) and the EU behind the spending levels seen in Asian countries (India and China). Even Japan, the 2nd largest R&D spender in the world, is now trailing the level of spending by China and India [6].

It is really hard to measure innovations, as its manifestation within the economy is larger and more complex than what one indicator or index can capture and reflect. Many aspects of technology creation, diffusion and human skills are hard to quantify.

Still in order to estimate nation's technological achievements and the level of innovative progress it is possible to use a great variety of indicators. The most frequently used ones are the following:

- The number of patents granted to residents (per million people), the number of new trademarks
- Receipts of royalty and license fees (US\$ per person)
- The number of researchers in R&D (per million people or per thousand employees)
- Population with tertiary education and youth education achievement level, new science and engineering (S&E) graduates per 1000 population
- Science and engineering degrees (% of all new degrees)
- % of firms with new-to-market product innovations (as % of all firms)
- Sales and exports of high technology products and many others

However, most of the indicators mentioned above describe only the quantitative side of innovation process, but not the efficiency of national R&D investment policy. Furthermore, some of these indicators are not representative due to considerable legislation differences among the countries (for example, low patenting activity in India and China is explained mostly by underdeveloped system of intellectual property rights' protection than by lack of innovations) [3]. That's why this research paper concentrates mainly on the share of high-technology exports as % of manufactured exports as the most representative indicator of competitive commercialization of national scientific researches (Y-variable).

While speaking about the factors of successful innovation policy, it is important to remember that there are no ideal models in complex economic systems, and each economic or social parameter is subjected to multitude of different impacts and factors. As regards national high-tech development, it is affected by such economic conditions as:

- International openness to trade and investments (assessed by such indicators as export and import ratio to GDP ($\text{Exp+Imp}/\text{GDP} \cdot 100\%$), the level of trade weighted average tariff)
- The commitment to market values and developed market economy infrastructure
- The accessibility to the venture financing for start-ups
- The competitiveness of national economy on the global arena
- The volume of R&D spending (in billions of \$)
- Industry innovation expenditures
- The influx of direct foreign investments and many others
- Public and private R&D expenditures (% of GDP)

Research and development (R&D) expenditure is one of the most widely used measures of the innovative efforts of firms and countries. Most surveys devoted to the technological achievements of the countries concentrate the attention mainly on the level of R&D intensity of this country as the main factor. But while R&D as a percent of GDP figures are bandied about as indicators of the strength of the national commitment to scientific research, they have relatively little meaning in terms of just how that investment contributes to the growth and welfare of the country.

The more important data are those that tell you who is providing the funding, who is doing the work, how the money is being spent, and what the priorities, thrusts, and

directions are. In brief, it is the internal structure of the R&D enterprise and the roles and interplays among the different sectors that have a bearing on the manner in which the investment in R&D has the desired societal benefit outcomes of economic security, improved health care, and the like. The R&D expenditure is generally broken down among 4 sectors: business enterprise, government, higher education and private non-profit institutions. In this research the share of business financed R&D was selected for thorough econometrical analysis (X variable).

2 Results

The basic hypothesis based on the preliminary insights into the statistical data suggests that, in order to implement effective and profitable “high-tech policy”, governments have to intensify the share of business sector in GERD (Gross Domestic Expenditures on R&D). But at the same time it is necessary to preserve the definite “government share” in R&D investments, as reduction of it up to certain extent gives the negative effect.

Figure 1 illustrates the average indications of high-technology exports (red line) and business expenditures on R&D (blue line) during the period 2000-2005 for 20 countries (the countries were arranged in order of business R&D share extension) [1], [3], [7].

We can see from the graph that the supposed rule is valid for the countries disposed in the range of 0-60 % share of business sources in R&D expenditures: the higher share of business sector in R&D means the higher indications of high-tech export.

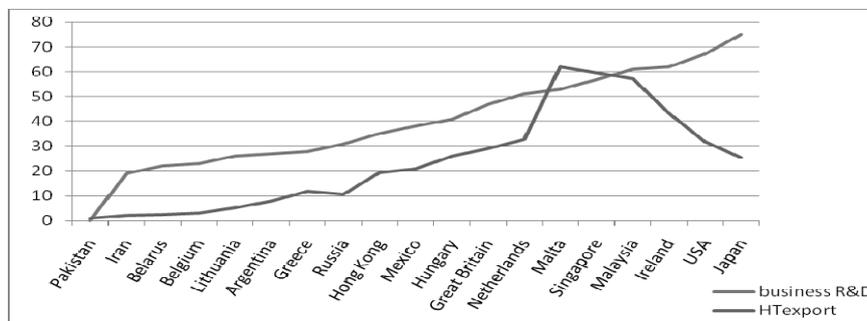


Fig. 1. The average indications of high-technology exports and business expenditures on R&D. Business R&D – % of Gross Domestic Expenditure on R&D financed by Business Sector; HTexport – % of high-technology export in total manufactured export

However we cannot fully rely on average indications, as each country has its own peculiarities, historic and geographic conditions and many other factors that can determine high-technological specialization of export. Moreover, here only 20 countries were taken into account.

That's why it is more interesting and important to examine the changes in the share of high-technology exports depending on the changes in the structure of R&D financing.

The Model description:

X ($\Delta\text{privR\&D}$) – shift in the share of business sector in R&D financing. Business R&D expenditures as % of total R&D expenditures – the indicator reflects the percentage of total investment in research and development originating from the business sector;

Y (ΔHTExp) – shift in the share of high-technology exports as % of total manufactured export during the period 2000-2005.

High technology export is exports of products with a high intensity of research and development. They include high-technology products such as those used in aerospace, computers, pharmaceuticals, scientific instruments and electrical machinery [4].

The statistical analysis of the data across 63 countries in the world for the period of 2000-2006 intended to reveal the correlation between changes in R&D expenditures structure and export structure. It will be studied linear regression.

According to the statistical analysis it was revealed the following general tendency: the share of the high-tech export increases in most of the countries surveyed along with the growth of business financed share in R&D investments.

The econometric model on the basis of the statistical data has the following outlook (figure 2):

$$\Delta\text{HTExp} = -1,8071 + 0,7129 * \Delta\text{privR\&D} \quad (R^2 = 0,636),$$

which means the increase of high-technology export share by nearly 0,7% if the share of private R&D expenditures grows by 1%.

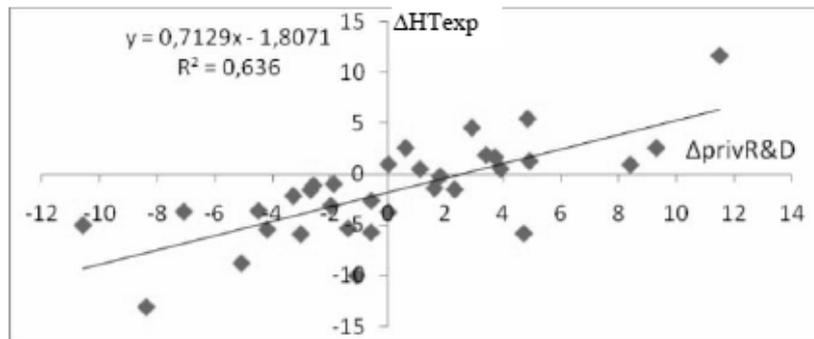


Fig. 2. The linear regression

With the purpose of further analysis the countries were classified into 3 categories:

The countries with traditionally high share of private sector in R&D (>60%). This group includes such countries as Belgium, Denmark, Finland, Germany, Ireland, Israel, Japan, China, Luxembourg, Switzerland, USA (figure 3).

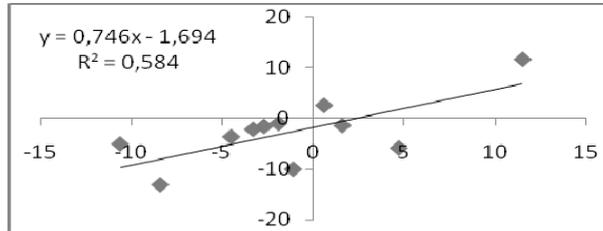


Fig. 3 The countries with high share of private sector in R&D

The countries with medium share of private R&D expenditures in the range from 40% to 60% are composed of such countries as Austria, Brazil, Croatia, Cuba, Czech Republic, France, Hungary, Netherlands, Spain, Great Britain (figure 4).

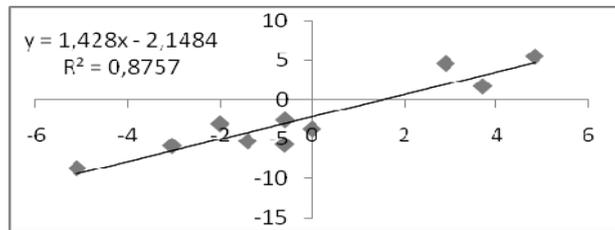


Fig. 4 The countries with medium share of private sector in R&D

The countries with traditionally low share of private sector in R&D (<40%) are such countries as Azerbaijan, Belarus, Bulgaria, India, Iran, Latvia, Pakistan, Poland, Portugal, Russia, Ukraine.

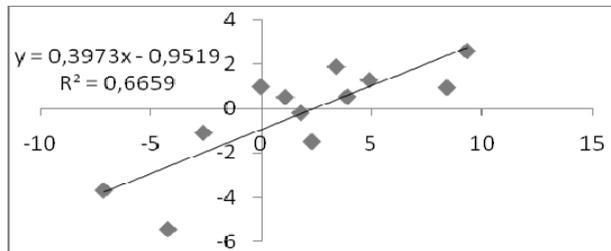


Fig. 5 The countries with low share of private sector in R&D

3 Conclusions

1. For 3 groups of countries (with different level of business expenditures) the trend line has the positive angle, which confirms the basic hypothesis. The peculiarity here is that the elasticity of high-tech exports to the private R&D investments is higher for the 2nd group of countries (where 40-60% of R&D is financed by business sector). It is the diapason in which the most drastic changes in export structure happen with the increase or decrease of business share in R&D investments.

Also it is important to highlight that the 3rd group of countries (with low participation of business sector) mainly has the tendency to declining share of high-tech export (most of the countries are located in the 3rd quadrant on the graph).

2. The share of business investments in R&D to the extent more than 80% may cause the decline in high-tech export. So the government expenditures are an important factor of accelerating the further R&D investments. According to the Harrod–Domar theory more investment leads to capital accumulation, which generates economic growth. Regarding the R&D investment policy, it is possible to make an assumption that expenditures of the government on R&D create the basis and indispensable minimum from which further R&D investment activity is multiplied.
3. Time lag according to statistical data is no more than 1 year.

Explanation of the results, received in the research.

The government is involved mainly in financing the fundamental investigations and basic research (basic research is experimental or theoretical work undertaken primarily to acquire new knowledge of the underlying foundation of phenomena and observable facts, without any particular application or use in view [2]), which implies low degree of commercializing of that kind of R&D.

The main commercial projects of the government in R&D sphere are concentrated in such fields as defense, healthcare, space programs, infrastructure. The governments of the developing countries invest in import-substituting industries, which also mean low level of transforming the financed R&D into high-technology export.

Finally, even in the cases when the government takes the lead in innovation financing and implements different governmental programs for innovative development, it cannot respond better to the changing market necessities and conditions, than private investors and companies that are interested to the maximum extend in the commercial success of their investigations. The governmental programs on the other side frequently tend just to expand the range of goods, but not the technological structure of industry and its qualitative parameters.

Policy recommendations.

It is important to focus on increasing efficiency in R&D spending rather than meeting a specific spending level. The efficiency and competitiveness of R&D investment policy, in its turn, can be achieved by expanding of the role of business sector in R&D financing up to 70-80% (it is important to use economic incentives such as tax exemptions, for example)

It is necessary to preserve the government spending on R&D within the level of 15-20% of the total expenditures, maintaining at the same time flexibility in allocating public R&D funds. The government should concentrate mainly in the basic researches sphere and accelerate in this way other fields of R&D investments.

References

1. OECD: Main Science and Technology Indicators, [http://www.oecd-ilibrary.org/science-and-technology/main-science-and-technology-indicators/volume-2008/issue-2_msti-v2008-2-en-fr;jsessionid=2obsve2k0wjt3.x-oecd-live-01\(2008\)](http://www.oecd-ilibrary.org/science-and-technology/main-science-and-technology-indicators/volume-2008/issue-2_msti-v2008-2-en-fr;jsessionid=2obsve2k0wjt3.x-oecd-live-01(2008))

2. OECD: Factbook 2008: Economic, Environmental and Social Statistics, http://www.oecd-ilibrary.org/economics/oecd-factbook-2008/world-population_factbook-2008-graph1-en (2008)
3. OECD: Science, Technology and Industry Scoreboard, http://www.oecd-ilibrary.org/content/book/sti_scoreboard-2009-en (2009)
4. UNDP: Human Development Report 2007/2008, <http://hdr.undp.org/en/reports/global/hdr2007-2008> (2007)
5. Battelle and R&D Magazine: Global R&D report 2008, <http://www.asiaing.com/2008-global-r-d-report.html> (2008)
6. Battelle and R&D Magazine 2010: global R&D funding forecast, <http://www.rdmag.com/topics/global-r-d-funding-forecast?page=2> (2010)
7. World Bank database, www.worldbank.org

Econometric Analysis on the Site “Lesson Pulse”

Alexander J. Weissblut

Kherson State University, 1, 40 rokiv Zhovtnya Street, 73000, Kherson, Ukraine

veits@ksu.ks.ua

Abstract. In this article the site “Lesson pulse” is considered, as the tool allowing the teacher to receive the objective information on a course and results of a lesson in a mode online. However adequate interpretation for results of such interrogations is impossible, while we will not separate true students from others. Besides, interpretation for results of interrogations and the decision-making, grounded on it, demands to realize, what exactly this concrete group means by clearness of an explanation, objectivity of the marks etc. For anonymous interrogations it means necessity of correlation and regression analysis for their results and an estimation of the statistical significance of the received results. So it means necessity of econometric analysis.

Keywords. Factor, statistical, econometric, analysis, correlation, decision-making

Key terms. Research, Management, Model, KnowledgeManagementProcess, KnowledgeManagementMethodology, MathematicalModeling

1 Introduction

In this article the site “Lesson pulse” is considered, as the tool allowing the teacher to receive the objective information on a course and results of a lesson in a mode online. It allows for the student or the pupil at any moment to react to a lesson course, having answered one or several questions, for example:

1. Is it interesting to you at a lesson?
2. Is it accessible (clear) an explanation?
3. Are you tired? Whether arranges you the rate of an explanation?
4. There are at you questions to the teacher?
5. Whether the marks are objective?

(Formulations of questions are defined by the teacher). As a result of an average of these responses the site produces on the monitor screen the data about a lesson state, its "pulse" in a mode online. At any moment the teacher can ask to answer all simultaneously such or more profound groups of questions (their examples are given below). So, he can measure the “lesson pulse” just at this moment. Such interrogations

do not demand computer auditorium by all means: they can be carried out on one tablet, and then results can be transferred to a site.

However adequate interpretation for results of such interrogations is impossible, while we will not separate true students, for which educational process is a considerable part of their life, from those, who would prefer to keep far away from it. Besides, interpretation for results of interrogations and the decision-making, grounded on it, demands to realize, what exactly this concrete group means by clearness of an explanation, lesson atmosphere, objectivity of the marks etc. For anonymous interrogations it means necessity of correlation and regression analysis for their results and an estimation of the statistical significance of the received results. So it means necessity of econometric analysis.

1. All groups of questions considered further have been chosen in result of "brainstorming" where students of fourth year study of the Faculty of physics, mathematics and informatics at the Kherson state university acted as experts. This expert interrogation has been constructed by a technique of "six hats of thinking" E. Bono [1], which provides the maximal openness and relaxedness of participants. In all cases the opinion has unanimously been expressed, that the given set of questions is full and fair.
2. Then students of specialities "physics", "mathematics", "informatics" and "program engineering" of the Kherson state university have been interviewed under such essential requisites. The respondents estimate each question from 0 (at firm "no") up to 10 (at firm "yes"). He arbitrarily sets a name of the folder containing his interrogation (i.e. his key). The volunteer – a participant of interrogation – collects all folders in one main folder and sorts them here (i.e. shuffles). Only after that the main folder was transferred to the teacher: this simple and open procedure guaranteed to participants anonymity of interrogation. Alternative and technically simpler variants are answers to the site and to a tablet: the variant choice is defined by a kind of interrogation and level of trust of an audience to the interviewing teacher.
3. Results of interrogation then are transferred to a site "Lesson pulse", which is realized in language PHP and uses database MySQL (see [2]). The queries realizing now on the site give out results of the econometric analysis of interrogation. They include the plural correlation analysis of factors and an estimation of the statistical importance of the received results with use of criteria Student and Fisher (see [3]). The site interface is oriented to the user, generally speaking, nothing knowing about the econometric analysis.

2 The Analysis of Interrogations on Results of Lesson and Feedback

Results of interrogation about lesson and interrogation Feedback are, of course, absolutely various depending on a lesson, a teacher, an audience etc. However the correla-

tion analysis of factors led to similar outcomes (at 20% a significance level by criterion of Student). Everywhere below we use the interrogations of 421 groups (speciality “mathematician”), having typical species (fig. 1).

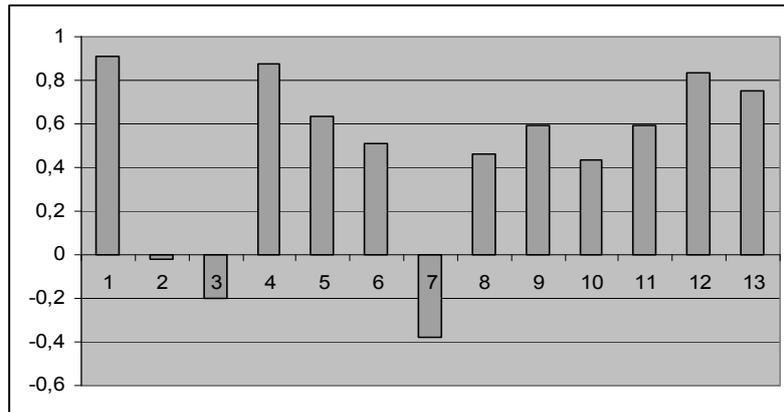


Fig. 1. Histogram for distribution of correlation coefficients Q1

Here is the histogram for distribution of correlation coefficients between answers to a question “Whether the lesson was pleasant to you?” and following factors:

1. Whether it is accessible (clear) an explanation?
2. Whether arranges you the rate of an explanation?
3. Are you tired at a lesson?
4. Lesson atmosphere: is it comfortable, is it pleasant to you at a lesson?
5. Is the statement filled enough by examples?
6. Objectivity of the marks, which have been put down at a lesson.
7. Are you having some questions to the teacher?
8. Do you still want a lesson on this theme?
9. Have you prepared for this lesson?
10. Are you intending to continue studying at home?
11. Accordance of a lesson to tasks of independent (home) work.
12. Is it interesting to you at a lesson?
13. Have you taken out something useful at a lesson or are sorry about spent time?

The most significant factors had appeared (in decreasing order) **1** (0.91), **4** (0.87), **12** (0.83), **13** (0.75) **5** (0.63), **9** and **11** (0.59). Objectivity of marks is only further (0.51) and inverse correlation – 0.39 for **7** specifies that for the majority the good lesson is such after which does not remain questions to the teacher (fig.2).

Here is the histogram for distribution of correlation coefficients between answers to a Feedback question “Whether the teacher is pleasant to you?” and following factors:

1. Whether lessons were pleasant to you?
2. Estimation by student of the knowledge received at lesson.
3. Is it accessible (clear) an explanation?

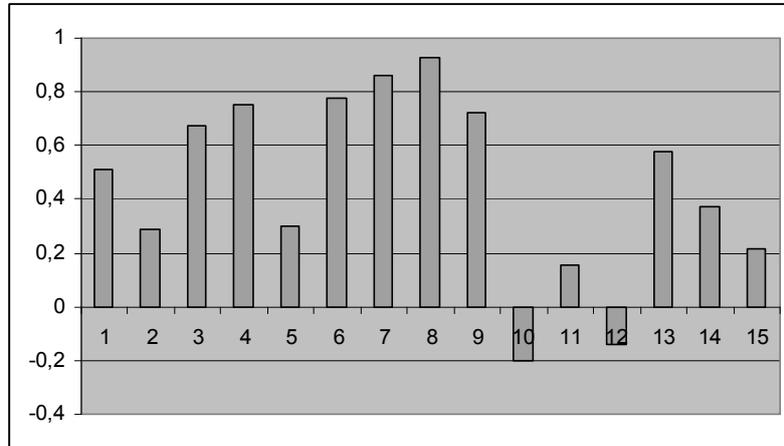


Fig. 2. Histogram for distribution of correlation coefficients Q2

4. How much are accessible (clear) and authentic answers to questions of students?
5. The explanation is filled enough by examples.
6. Using of various approaches at training.
7. Whether the teacher aspires to interest, motivate students?
8. Lessons atmosphere: is it comfortable, is it pleasant to you at a lesson?
9. Availability of the teacher, his inclination to listen to students, to conduct discussions with them.
10. Knowledge of a subject by the teacher.
11. Insistence (regular and frequent enough control of knowledge).
12. Punctuality (comes in time at lessons).
13. Possession of an audience (students do not sleep and do not make too much noise at lessons).
14. Objectivity in estimation of the student by the teacher. Whether criteria of estimation in all subgroups are identical?
15. Accordance of a lesson to control tasks.

The most significant factors appear (in decreasing order) **8** (0.92), **7** (0.85), **6** (0.775), **4** (0.75), **9** (0.72), **3** (0.675), **13** (0.58).

Only further with factor of correlation 0.51 follows 1 - whether lessons were pleasant to you. And major factors of estimations of the teacher and lesson are considerably differing. Further the histogram of differences between factors of correlation for questions “Whether the teacher is pleasant to you?” and “Whether lessons were pleasant to you?” is resulted.

The factors much more essential at an estimation of a teacher, than a lesson are 6 (using of various approaches at training) and 7 (whether the teacher aspires to interest, motivate students). On the contrary, at an estimation of a lesson it is much more essential factors 14 (accordance of a lesson to control tasks) and 10 – insistence (regular and frequent enough control of knowledge): probably, according to students, insistence it is good for lesson and it is not so good for the teacher.

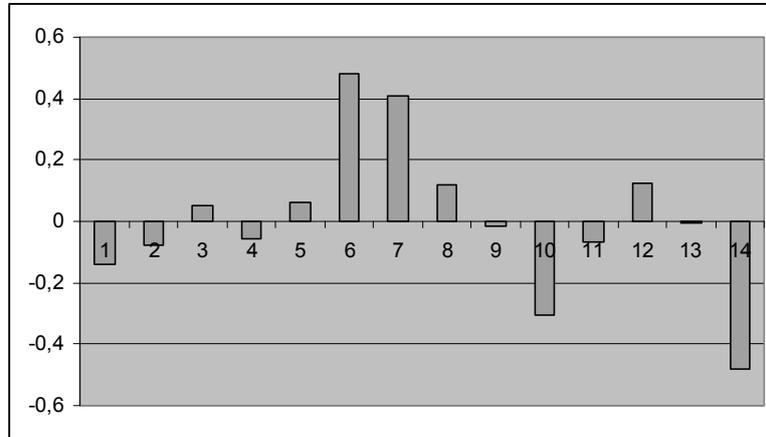


Fig. 3. Histogram for distribution of correlation coefficients Q3

Certainly, the correlation matrix contains decomposition on factors also for each of 15 questions. So it is found out that 5 (the explanation is filled enough by examples) is most closely connected with 15 (accordance of a lesson to control tasks); 3 (are you tired at a lesson) with 7 (presence of questions to the teacher); 13 (possession of an audience) with 14 (objectivity in estimation of the student).

It is interesting to compare 12 (is it interesting to you at a lesson) with 13 (have you taken out something useful at a lesson) from interrogation about results of the lesson (fig.4).

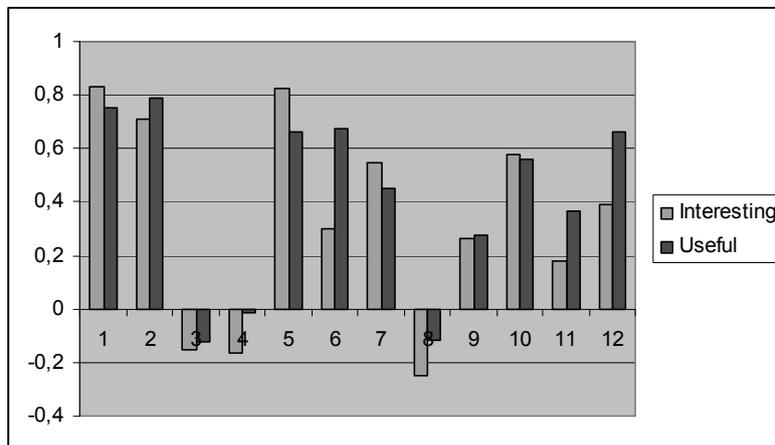


Fig. 4. Histogram for distribution of correlation coefficients Q4

As we see, from the student’s point of view, it is interesting and it is useful is not the same. So 4 (lesson atmosphere) correlates with the factor interesting much more, while factor 5 (is the statement filled enough by examples) with 11 (accordance of a lesson to tasks of independent (home) work).

3 The Analysis of Interrogations about the Factors Influencing a Lesson

Unlike interrogations about results of lesson and Feedback results of interrogations about factors of influence on a lesson course are close enough in different groups. The histogram for distribution of interrogation requisites on the relation to lesson (in 421 group) is below (fig. 5).

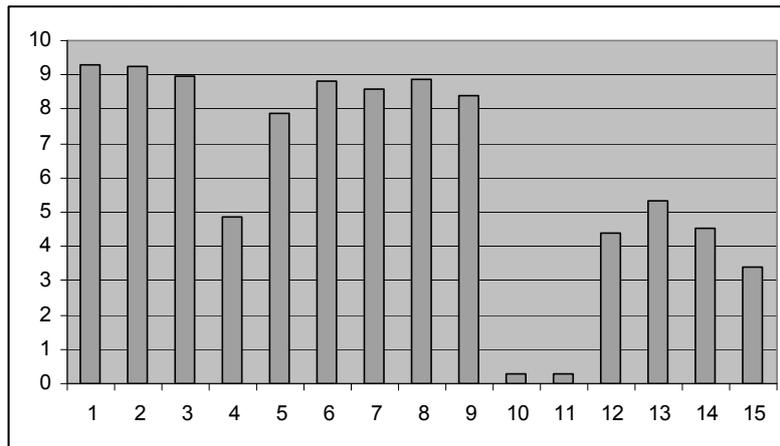


Fig. 5. Histogram for distribution of correlation coefficients Q5

Here:

1. Is study pleasant to you? Is it interesting to you?
2. Whether you believe that education is “road to the future”?
3. Is your speciality pleasant to you?
4. Is the program of training for your speciality satisfying you?
5. Whether satisfies you teaching level at the university?
6. Whether on own will you have chosen university and a speciality?
7. Would you like to change the speciality or to receive additional higher education?
8. Whether your attendance of lessons is regular?
9. Do you regularly prepare homework?
10. Whether there were at you conflicts to teachers?
11. Were you afraid of an exception of university?
12. Do you wish to take part in scientific work, in Olympiads on your speciality?
13. Whether often to you fellow students address for the help in lessons?
14. Do you wish to enter postgraduate study after training end?
15. What’s the time you spend for preparation for lessons (on the average hours per day)?

And further similar results of interrogation on external factors (fig. 6):

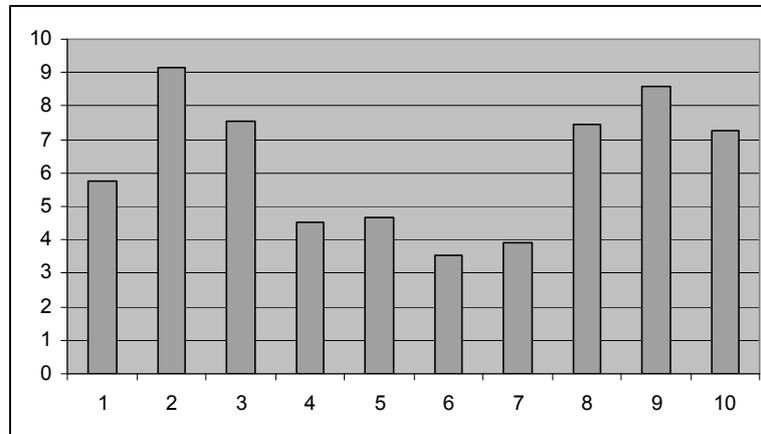


Fig. 6. Histogram for distribution of correlation coefficients Q6

Here:

1. Close dialogue with teachers.
2. Accessibility of the Internet at university.
3. Readiness of an auditorium for a lesson (working capacity of projectors, computers, the software; comfort of an auditorium).
4. Presence of enough points for the centralized feeding.
5. Accessibility of contacts to the future employers.
6. Accessibility of summer improvement.
7. Participation in scientific work.
8. Teaching level at the university.

In a correlation matrix under all these factors there are only few factors which correlations are close to 1. These are factors:

1. Whether your attendance of lessons is regular with factors
 - (a) do you regularly prepare homework (0.87)
 - (b) teaching level at the university (0.84)
 - (c) participation in scientific work (0.63)
 - (d) have you prepared for this lesson (0.59)
 - (e) accessibility of summer improvement (− 0.5)
2. Do you regularly prepare homework with factors
 - (a) whether your attendance of lessons is regular (0.87)
 - (b) teaching level at the university (0.815)
 - (c) have you prepared for this lesson (0.66)
 - (d) participation in scientific work (0.56)
 - (e) accessibility of summer improvement (− 0.52)

3. Teaching level at the university with factors

- (a) whether your attendance of lessons is regular (0.843)
- (b) do you regularly prepare homework (0.815)
- (c) whether on own will you have chosen university and a speciality (0.65)
- (d) have you prepared for this lesson (0.59)
- (e) participation in scientific work (0.56)
- (f) accessibility of summer improvement (– 0.55)

Besides them correlation factors above 0.7 appear still only twice: between factors *whether there were at you conflicts to teachers* and *were you afraid of an exception of university* (0.85); and between factors *participation in scientific work* and *is the program of training for your speciality satisfying you* (0.74). Occurrence in such line the factor *teaching level at the university* is, probably, the best compliment for Faculty of physics, mathematics and informatics of the Kherson state university for all its history. Our main task is to use the mental orientation, fixed thus in the correlation analysis of factors, for separating true students, for which educational process is a considerable part of their life, from those, who would prefer to keep far away from it. Using already cited data and the following table 1

Table 1. Correlation analysis of factors

Factor	Average value	Root-mean-square deviations
Teaching level at the university	702	2.17
Regularly attendance of lessons	8.85	2.3
Regularly prepare homework	8.4	2.6

We choose as a differentiating sign between groups the factor *regularly prepare homework*. In this case mutual correlations of defining sign are closer to 1; and the dispersion is more, that testifies about more variability of respondents under this factor. Besides, among others selected it more corresponds to such sign on common sense.

4 Results of Interrogations about Lesson and Feedback on Subgroups

To the selected differentiating sign among 20 respondents of group 421 the 12 participants is allocated, who for a question *do you regularly prepare homework* have answered with 10 or 9 points. The additional subgroup consists of 8 respondents. Whether there correspond such subgroups to required division into true students and the others? Below there is the histogram for average results of interrogation about lesson on the allocated subgroups (fig. 7).

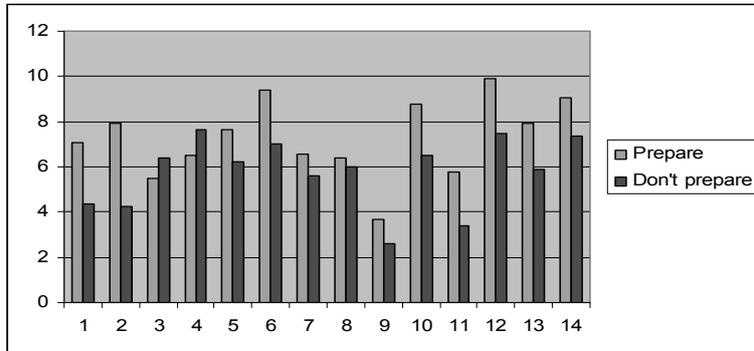


Fig. 7. Histogram for distribution of correlation coefficients Q7

So, the factors considerably different in subgroups (in decreasing order of modules of differences between average values in subgroups) are:

- 2 Whether it is accessible (clear) an explanation? $(7.92 - 4.25 = 3.67)$
- 1 Whether the lesson was pleasant to you? $(7.1 - 4.38 = 2.72)$
- 12 Accordance of a lesson to tasks of independent (home) work. $(9.91 - 7.5 = 2.41)$
- 6 Is the statement filled enough by examples? $(9.41 - 7 = 2.41)$
- 10 Have you prepared for this lesson? $(8.75 - 6.5 = 2.25)$
- 13 Is it interesting to you at a lesson? $(7.92 - 5.87 = 2.05)$
- 14 Have you taken out something useful at a lesson? $(9.1 - 7.4 = 1.7)$
- 5 Lesson atmosphere $(7.66 - 1.25 = 1.41)$
- 9 Do you still want a lesson on this theme? $(3.66 - 2.65 = 1.01)$

The averages of additional group are more only twice, there are:

- 4 Are you tired at a lesson? $(6.5 - 7.62 = -1.12)$
- 3 Whether arranges you the rate of an explanation? $(5.5 - 6.37 = -0.87)$

Last result is strange at first sight, but steady for all groups and it is easy to explain this phenomenon psychologically: as less adjusted the student for study, the more he would like acceleration, faster course of time.

Further there are similar results for interrogation Feedback (fig. 8).

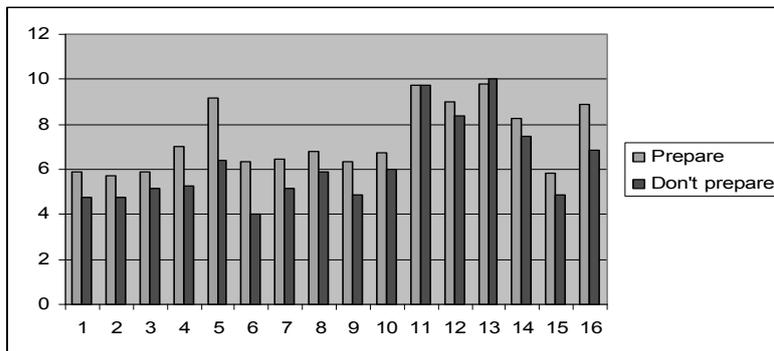


Fig. 8. Histogram for distribution of correlation coefficients Q8

Here the factors considerably different in subgroups are:

- 5 The explanation is filled enough by examples (9.17 – 6.37 = 2.8)
- 6 Using of various approaches at training (6.36 – 4 = 2.36)
- 16 Accordance of a lesson to control tasks (8.9 – 6.87 = 2.03)
- 4 How much are accessible (clear) and authentic answers? (7 – 5.25 = 1.75)
- 9 Lesson atmosphere (6.36 – 4.85 = 1.51)

The obtained data corresponds to a hypothesis about required division into groups, anyway they don’t contradict it.

5 The Latent Division in Group

The site “Lesson pulse” offers also division of group into classes with a given value of mutual correlation: between two respondents from one class it is possible to find a chain of respondents of this class so, that the correlations of answers between consecutive respondents of this chain not less than the given value. Such division into subgroups allows finding out distinctions in the group, which is not appreciable directly.

At mental interrogation about factors of influence on lesson and the set minimum level of mutual correlation 0,6 in test group 421 splitting into 3 classes has turned out: from 4, from 5 and from basic subgroup of 11 respondents. Let’s compare averages of the basic class to averages of the first and second subgroups under those factors in which appreciable differences have come to light.

1. Is the program of training for your speciality satisfying you?
2. Would you like to change the speciality or to receive additional higher education?
3. Do you wish to take part in scientific work, in Olympiads on your speciality?
4. Do you wish to enter postgraduate study after training end?
5. Participation in scientific work.
6. Readiness of an auditorium for a lesson.
7. Accessibility of summer improvement.
8. Accessibility of contacts to the future employers.

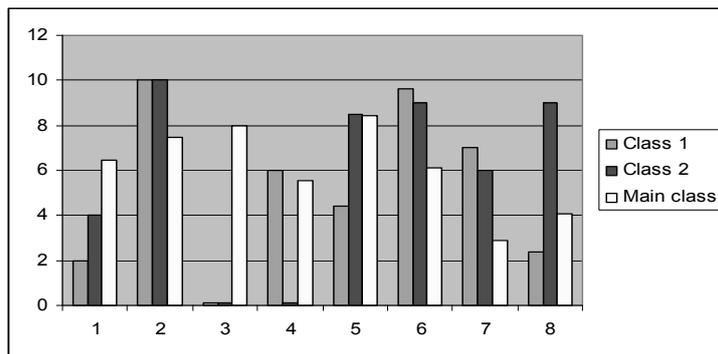


Fig. 9. Histogram for distribution of correlation coefficients Q9

Respondents from classes 1 and 2 much less than the basic group are satisfied by the program of training of the speciality (point 1). They would like to change the speciality or to receive additional higher education essentially more than the basic group (point 2). Their difference is the sharpest comes to light in point 3: unlike the basic group they at all do not wish to take part in scientific work or in the Olympiads on the speciality. So, apparently, the speciality for them has lost now appeal. Respondents from a class 2 does not interest in the postgraduate study (point 4), however they do not against scientific work (point 5). The main thing, they have the most interest in contacts to employers (point 8). Apparently, it is search of the application out of the speciality. Respondents from a class 1 are focused differently: they have a little interest in scientific work and employers (points 5 and 8), but they wish to enter postgraduate study (point 4).

References

1. De Bono, E.: *Six Thinking Hats*. Penguins Books (1997)
2. PHP Book, <http://www.phpreferencebook.com> (2012)
3. Hansen, B. E.: *Econometrics*. Textbook, <http://www.ssc.wisc.edu> (2012)

Decision Supporting Procedure for Strategic Planning: DEA Implementation for Regional Economy Efficiency Estimation

Karine Mesropyan¹

¹ Institute of Socio-Economic and Humanitarian Researches of Southern
Scientific Center of RAS, 41 Chekhova prospect, 344006 Rostov-on-Don, Russia

karineemesropyan@gmail.com

Abstract. The algorithm of decision supporting procedure based on Data Envelopment Analysis (DEA) along with the Malmquist Productivity Index is suggested in the paper. The procedure's core consists of evaluations complex for preliminary data processing and adjustment as well as creating of analytic materials in the field of regional strategy planning. The crucial study issue is to define boundaries of DEA applicability in this field and to eliminate DEA shortcoming, such as the scores dependence on a set of inputs and outputs. The efficiency scores of Russian regional agrarian sector are obtained in order to verify the procedure and add knowledge to current indicators' systems of regional economic efficiency by improving approach objectiveness. It is shown how obtained results can be applied in the strategic planning to increase effectiveness of state regional policy activity.

Keywords. Efficiency, Malmquist Index, Data Envelopment Analysis, Region, Procedure, Strategy Planning

Key terms. DecisionMaking, MathematicalModel, Methodology, Development, Management

1 Introduction

Theoretical model of this study is based on the Pareto-Koopmans concept (1951) [1]. System technology is efficient by Pareto-Koopmans if and only if the object does not have an opportunity to improve its resource (input) or product (output) without sacrificing some other input or output. Charnes et al. (1978) have proposed Data Envelopment Analysis (DEA) based on this concept of efficiency that was combined operational research tools within works of Koopmans (1951) and Farrell (1957) [2, 3].

DEA is a non-parametric frontier approach for comparative efficiency measurement in which a set of similar objects with multiple inputs and outputs is analyzed.

The aim of this study is to suggest the procedure for providing strategy planning by analytical reports based on DEA scores implementation.

It is obvious that productivity analysis by DEA has at least three current issues. The first is to define a set of objects which will be compared in the study. The second is to formulate convenient conditions for concrete models' modifications using. The third issue is to improve discrimination capability. Therefore, efficiency assessment procedure by DEA is primarily based on the following grounds: the formation of objects' set to be compared, identification of inputs and outputs, and model selection.

Taking into account issues mentioned above, it is necessary to adapt basic DEA models and its implementation. Furthermore, DEA procedure is considered as a core of the evaluation of regional economy efficiency scores.

This paper consists of five parts. We state the main issues in this, first, part. This study's background is presented in the second part. Part 3 deals to description of suggested evaluation procedure. The applying of investigation of the Russian regions agrarian sector to management tasks by using the procedure is reported in the part 4. We make conclusions in the last part.

2 Theoretical and Methodological Background

DEA application has a big number of advantages. First of all, a calculation of an integrated assessment is produced for each region reflecting the efficiency of input factors using for output products. Besides, the Pareto-optimal set of efficient regions in the multidimensional space of inputs and outputs is being obtained. Secondly, it is unnecessary to attract an expert knowledge in a priori assignment of weights for variables corresponding to inputs and outputs. Despite of this, using of additional data on region external factors is helpful for creating the right model. Thirdly, it is very important that there are no restrictions on the functional form of the relation between inputs and outputs.

The study is carrying out with the hypothesis that DEA implementation needs the formal procedure in order to obtain stable scores and apply research results to analytic background of current regional strategic planning.

The multilateral and penetrating analysis of DEA possibilities and its application's restrictions are presented in Dyson et al. (2001), Cook and Seiford (2009) [4, 5]. Along with these works there are reviews of this method application, for instance, in papers of Avkiran and Parker (2010), Liu et al. (2012) [6, 7]. The common bases productivity measurement presented in Caves et al. (1982) [8].

The application possibilities of Malmquist Productivity Index in different intertemporal comparisons are described in the research of Färe and Grosskopf (1996) [9]. Tsuneyoshi et al. (2012) used Malmquist Index for the comparative analysis of 97 countries calculated by DEA models for period 1981-2004 [10]. Yamamura and Shin (2008) determined the nature of inequality impact on capital accumulation and growth performance by evaluation DEA indexes from 1965 to 1990 [11].

According to review presented in [12], although there are a DEA advantages, the general method's shortcoming is considered as crucial because the scores signifi-

cantly depend on a set of inputs and outputs. This study suggests the special procedure for DEA implementation for the needs of regional strategic planning. It is a result of attempting to eliminate the mentioned DEA drawback and provide the decision process of strategic planning by analytic materials. Golany and Roll (1989), Emrouznejad and De Witte (2010) offered procedures of DEA application which are very useful for common case [13, 14]. This study based on results of these works.

3 Efficiency Estimation Procedure

Different levels of the regional economy scale and the return to scale effect are considered as a reason of inequality between regional output performances. That is why the model with variable return to scale is suggested for this study. This model was introduced by Banker et al. (1984).

Data for a research by DEA is presented by a number of indicators in form of the matrix of inputs $X^j = \{x_{ij}^j\}$ and matrix of outputs $Y^j = \{y_{kj}^j\}$. The efficiency criterion for a multidimensional assessment of an object is to assign some input and output parameters for all objects, some weights and then to calculate and maximize the ratio for each object:

$$\theta_j = \frac{\sum_{k=1}^s u_k y_{kj}^j}{\sum_{i=1}^m w_i x_{ij}^j} \tag{1}$$

where:

- j – index of the estimated production facility, $j=1, \dots, n$;
- x_{ij} – matrix of input parameters that reflect the system resources, $i=1, \dots, m$;
- y_{kj} – matrix of outputs which reflect the products of system, $k=1, \dots, s$;
- u_k, w_k – weights for outputs/inputs.

According to the DEA framework, this function should be maximized under restrictions for all objects:

$$\frac{\sum_{k=1}^s u_k y_{kj}^j}{\sum_{i=1}^m w_i x_{ij}^j} \leq 1 \quad \forall j = 1, \dots, n; \quad u_k \geq 0, w_i \geq 0 \tag{2}$$

The Malmquist Productivity Index is calculated using such DEA efficiency scores for evaluation of total factor productivity change:

$$M_j^{t+1}(x^{t+1}, y^{t+1}, x^t, y^t) = \left[\frac{\theta_j^t(x^t, y^t) \theta_j^{t+1}([x]^{t+1}, y^t)}{\theta_j^t([x]^{t+1}, y^{t+1}) \theta_j^{t+1}([x]^{t+1}, y^{t+1})} \right]^{\frac{1}{2}} \tag{3}$$

The suggested procedure for regional efficiency assessment has the complex of procedures for preliminary data processing and adjustment (fig.1).

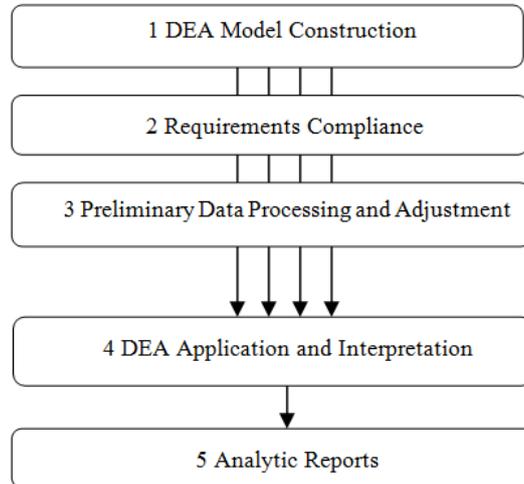


Fig. 1. Five stages of Efficiency Estimation Procedure

The dual linear program model for evaluation the criterion given above is:

$$\begin{aligned}
 \min_{\eta, \lambda} \eta \quad \text{subject to} \quad & y_{ku} - \sum_{j=1}^n \lambda_j y_{kj} \geq 0, \quad k = 1, \dots, s, \\
 & \eta x_{i0} - \sum_{j=1}^n \lambda_j x_{ij} \geq 0 \quad i = 1, \dots, m, \quad \sum_{j=1}^n \lambda_j = 1, \quad \lambda_j \geq 0, j = 1, \dots, n
 \end{aligned}
 \tag{4}$$

where:

η – comparative efficiency score of region j ($j=1 \dots, n$);

λ_j – dual model variables.

If $\eta < 1$ then the region belongs to inefficient set, otherwise ($\eta = 1$) it is a part of Pareto set.

According to procedure carrying out, the result of all evaluations using (3), (4) is a set of regional types by dynamics character.

Generally, the result of DEA application is the set of scores also which shows the ways of comparative efficiency improvement for each inefficient region. Nevertheless, this issue is not treated in this procedure because it requires the special attention and investigation due to its complexity.

4 Evaluation by Using Procedure

We examined the issue of inequality of regional economy performance for the period from 2008 to 2010. Federal State Statistics Data is used from www.gks.ru [15]. We

evolved a set of indicators that can be used in a broader context in order to identify factors influencing on a regional underdevelopment.

Stage 1. The set of indicators for models consists of resources and results of regional agrarian sector performance. Indicators are reported in Table 1.

Stage 2. Set of model’s variables defined this way: five resources are taken as inputs while three results are taken as outputs. The volume of region population is considered as the special variable for the set’s normalization.

Table 1. The set of regional performance indicators

Type	Indicator
Resources	number of cattle, thousand heads (x_1)
	organizations acreage under crops, ha (x_2)
	average number of employees, thousand people (x_3)
	power capacity, thousand horsepower (x_4)
	equipment park (tractors), units (x_5)
Results	gross grain yield, thousand tons (y_1)
	production of milk, thousand tons (y_2)
	production of livestock and poultry, thousand tons (y_3)
Variable for Normalization	volume of region population, thousand people

Stage 3. Next, the homogeneity of conditions was checked for all agrarian regional systems, and asymmetry of land’s quality founded out. The input called “organizations acreage under crops” is adjusted by the coefficient of cadastral value of agricultural land. The rule of ratio of variables’ number and objects’ number is kept, therefore, modeling is made for 53 quite similar agrarian Russian regions.

Additional restriction to weights is used in order to improve the discrimination capability of the model. Direct restriction on the ratio of the quantity of employees and the power capacity presented by the following ratios:

$$\alpha \leq \frac{x_3}{x_4} \leq \beta \tag{5}$$

$$\alpha \min(\text{price}_{x_2}) = \frac{\square}{\min(\text{price}(\square_{x_4}))} \tag{6}$$

$$\beta \max(\text{price}_{x_2}) = \frac{\square}{\max(\text{price}(\square_{x_4}))} \tag{7}$$

where:

$price_{x_3}$ – regions' average monthly salary;

$price_{x_4}$ – regions' average energy price;

x_3, x_4 – inputs which are taken in the normalized forms according to the second stage.

Stage 4. According to (3) and (4), the calculation cycle is done, and obtained scores are insensitive regarding the model parameters changes. It was approved by decreasing of the set of analyzed objects. Besides, the Malmquist Indices values are similar to current expert opinion on the character of current tendencies of technological progress changes in the industry for analyzed period.

Stage 5. The quantitative scores combined with qualitative evaluation of risks and conditions of regional development allow finding out the regions taxonomy by using obtained knowledge on type of efficiency dynamics. The procedure has conducted from the first to the fifth stage given in Fig.1.

The most significant agrarian regions of Russia are located on the Southern territory which consists of two state districts, namely Southern Federal District and Northern Caucasus Federal District. There are two strategies for these regions: Southern Federal District Strategy and Northern Caucasus Federal District Strategy [16, 17]. Obtained results can be part of analytic reports of these policy development documents (tables 2-3). Development scores are presented for period 2008 - 2010 in the tables. The indicator is equal to «+»/«-» in the case of positive/negative dynamics.

Table 2. Efficiency Scores for the Southern Federal District Strategy

Region	Development Score	Region Type
Republic of Adygeya	++	Stable Growth
Republic of Kalmykiya	++	Stable Growth
Krasnodar Region	-+	Unstable Decline
Astrakhan Region	+-	Unstable Growth
Volgograd Region	++	Stable Growth
Rostov Region	+-	Unstable Growth

Table 3. Efficiency Scores for the Northern Caucasus Federal District Strategy

Region	Development Score	Region Type
Daghestan	--	Stable Decline
The Ingush Republic	++	Stable Growth
Republic of Kabardino-Balkariya	--	Stable Decline
Republic of Karachaevo-Cherkesiya	--	Stable Decline
Republic of Northern Osetia Alaniya	-+	Unstable Decline
The Chechen Republic	--	Stable Decline
Stavropol Region	++	Stable Growth

Although analyzed period covers the crisis years, the agrarian production of the South of Russia shows the reserve of stability. Besides, it is brought out that the Southern regions belong to Pareto-Efficient set of Russian regions.

Thus, only 4 regions among 13 of the South are estimated as having the stable decline. The economic development opportunities of this regions are significant, nevertheless the considerable potential of regions is not using.

Such indicators' further analysis can be used for adjustment of scenario data tasks in the field of regional development foresight and strategic planning. The obtained results also can be suitable for equalization policy design in order to steady the level of regional efficiency during long-term period. In addition to this, it is important that risks, conditions and possible consequences of the policy should be assumed for each scenario of regional development.

5 Conclusions

The verification of suggested procedure along with the DEA model demonstrates the positive results that approve the possibility of the procedure application to prospective studies in the field of production analysis as well as strategic management.

As it was shown, the obtained results can be part of the quantitative investigations for the current strategies of development policy. In addition to this, the development scores can be used together with regional development risks and opportunities analysis, indicators of economical efficiency, such as gross domestic product per capita, enterprises profitability, etc. Thus, obtained scores will add knowledge to current indicators' systems of regional economic efficiency and improve approach objectiveness and effectiveness of state regional policy activity.

This article is conducted within Program of the Presidium of Russian Academy of Sciences № 32 "Fundamental Issues of Polyethnic Region Modernization in Terms of Tensions Growth".

References

1. Koopmans, T. C.: An Analysis of Production as an Efficient Combination of Activities. Activity Analysis of Production and Allocation. Cowless Commission for Research in Economics. Monograph No. 13, New York: Wiley, pp. 15–32 (1951)
2. Farrell, M. J.: The Measurement of Productive Efficiency. J. of the Royal Statistical Society, Series A (General), Part III. 120, 253–281 (1957)
3. Charnes, A., Cooper, W. W., Rhodes, E.: Measuring the Efficiency of Decision Making Units. European J. of Operational Research, 2, 429–444 (1978)
4. Cook, W. D., Seiford, L. M.: Data Envelopment Analysis (DEA) – Thirty Years On. European J. of Operational Research, 192, 1–17 (2009)
5. Dyson, R. G., Allen, R., Camanho A. S., Podinovski V. V., Sarrico C. S., Shale E. A.: Pitfalls and Protocols in DEA. European J. of Operational Research, 132, 245–259 (2001)
6. Avkiran, N. K., Parker, B. R.: Pushing the DEA Research Envelope. Socio-Economic Planning Sciences, 44, 1–7 (2010)

7. Liu, J. S., Lu, L. Y. Y., Lu, W.-M., Lin, B. J. Y.: Data Envelopment Analysis 1978–2010: A Citation-Based Literature Survey. *Omega* (2012)
8. Caves, D. W., Christensen, L.R., Diewert, W.E.: The Economic Theory of Index Numbers and the Measurement of Inputs, Outputs and Productivity. *Econometrica*, 50(6), 1393–1414 (1982)
9. Färe, R., Grosskopf, S.: *Intertemporal Production Frontiers: With Dynamic DEA*. Kluwer Academic, Boston, MA (1996)
10. Tsuneyoshi, T., Hashimoto, A., Haneda, S.: Quantitative Evaluation of Nation Stability. *J. of Policy Modeling*, 34, 132–154 (2012)
11. Yamamura, E., Shin, I.: Effects of Income Inequality on Growth through Efficiency Improvement and Capital Accumulation. MPRA Paper No. 10220, <http://mpa.ub.uni-muenchen.de/10220/> (2008)
12. Mesropyan, K., Goryushina, E.: Economic Heterogeneity and Political Instability: Experience and Prospects for Cross-Country Comparisons. *The Region Economy: Problems, Findings, Prospects*, Issue 13. OON RAN, ISERH SSC RAS, Volgograd, 58–66 (2012) (in Russian)
13. Emrouznejad, A., De Witte, K.: COOPER-Framework: A Unified Process for Non-Parametric Projects. *European J. of Operational Research*, 207(3), 1573–1586 (2010)
14. Golany, B., Roll, Y.: An Application Procedure for DEA. *Omega*, 1(3), 237–250 (1989)
15. Federal State Statistics Data, www.gks.ru (in Russian)
16. Southern Federal District Strategy, <http://www.minregion.ru> (in Russian)
17. Northern Caucasus Federal District Strategy, <http://www.minregion.ru> (in Russian)

Applying of Fuzzy Logic Modeling for the Assessment of ERP Projects Efficiency

Andriy Semenyuk

Lviv Academy of Commerce, Tugan-Baranovskoogo. 10, 79005 Lviv, Ukraine

andriy.semenyuk@gmail.com

Abstract. ERP software is one of costly and crucial projects for business investment. It is known that nowadays Enterprises can rarely afford to implement long-term projects, in most cases the duration of implementation varies from 3-4 months (automation of individual store retail chain) and 1-1.5 years when it comes to big projects. Only successful combination of analytical tools and methodologies will allow the project to realize and implement ERP-solutions for commercial enterprises on time and according to set business requirements. This paper proposes a practical assessment model which applies both the fuzzy analytic logic approach and the Expert Judgment method to evaluate whether the ERP software implementation project has succeeded or not.

Keywords. Modeling, Efficiency, Project, Information Technology, Implementation, Enterprise, ERP

Key terms. Model, Methodology, Process, Management, Approach

1 Introduction

Today it is especially important for Ukrainian Enterprises to be capable to analyze all the environmental aspects within they operate and be able to plan all needed resources with the most possible accuracy. To gain such competitive advantages within changing economy and unstable markets, for particular Enterprise is not enough just to have the most modern production lines or good educated personnel it also requires to possess some advanced technologies and modern information management systems that will quickly allow to react and adapt all the further changes. That is why more and more Enterprises in Ukraine from different sectors of economy are choosing to implement the Enterprise Resource Planning (ERP) system.

ERP plays an important role to integrate organization's information and functions in all areas of enterprise activities and within the variety of departments and finally results in successful operation on the market. However ERP implementation as a project itself is costly and time consuming, it also can lead to loss of many valuable resources of the company in case of wrong approached and not efficient way of imple-

mentation. So it is critically important for the Enterprises to understand and clearly realize all the value achieved from ERP initiative [2].

Many factors are essential in determining the efficiency of the ERP projects. Since most of these factors are qualitative and relations between them are very complicated, determining their exact quantitative values is quiet difficult. Using the combined methods of Expert Judgment and Fuzzy logic can be helpful to simplify the calculations and finally leads to a more precise result to determine generalized efficiency value of the implemented ERP-project. In this research, we intended to gather optimal KPIs values from different Enterprise activities and departments based on Experts Judgment data and design a combined Fuzzy Model to assess the efficiency for ERP-project [4].

2 Problem Statement and Goals of the Paper

The most common aspects and issues related to ERP systems development and implementation methods of ERP-projects seems to be widely discovered observed and investigated in works of many foreign as well as Ukrainian researches and scientists [1], [2], [4], [5]. However the problems related with particular to the efficiency of such projects are not enough covered. In view of this subject of the research is still topical.

The main aim of the paper is to present a model for evaluating the effectiveness of ERP-project implemented on the markets of underdeveloped economic systems with involves a combination of fuzzy logic and expert judgment methods. Because the initial data for measuring the effectiveness of ERP-project is mostly inaccurate and variable so the use of fuzzy logic techniques to enhance the data gathered from the expert is really feasible here.

3 Proposed Model and Approach

In order to develop the model for the assessment of ERP-project efficiency first of all it was conducted a set of related Expert Judgment questionnaires sessions, for the proposed assessment methods and optimal performance indicators or key performance indicators (KPI) values, in other words KPI is a type of performance measurement. An organization may use KPIs to evaluate its success, or to evaluate the success of a particular activity in which it is engaged. Sometimes success is defined in terms of making progress toward strategic goals, but often success is simply the repeated, periodic achievement of some level of operational goal (e.g. zero defects, 10/10 customer satisfaction, etc. Accordingly, choosing the right KPIs relies upon a good understanding of what is important to the organization [9]. The basics input data of our research are KPIs of ERP-projects that were received from managers of ERP-projects, and classified according to the criterion of "trend change".

Determination of further scope of KPIs and the major ERP success factors also is based on variety international consulting agencies reports KPIs values, and ERP-projects

statistics obtained from such reports was additionally verified with the involved experts [6], [7], [8].

As the result it has been allocated four main groups of indicators: 1) X - performance "increase group of KPIs" (the actual value of which increased for the Enterprise after ERP implementation), 2) Y - values "reduce group of KPIs" (the value of which decreased for the Enterprise after ERP implementation) 3) W – project financial and investments indicators. 4) Z - Generally optimized qualitative KPIs for different process aspects within Enterprise.

To be able assess the effectiveness of ERP-project we developed a structural combined model (depicted on Fig. 1). The model contains of methodological approaches and structure of key performance indicators to determine the effectiveness of ERP-project. Summarized decision tree inference, which is presented at the bottom of the model, reflects the hierarchy of input variables.

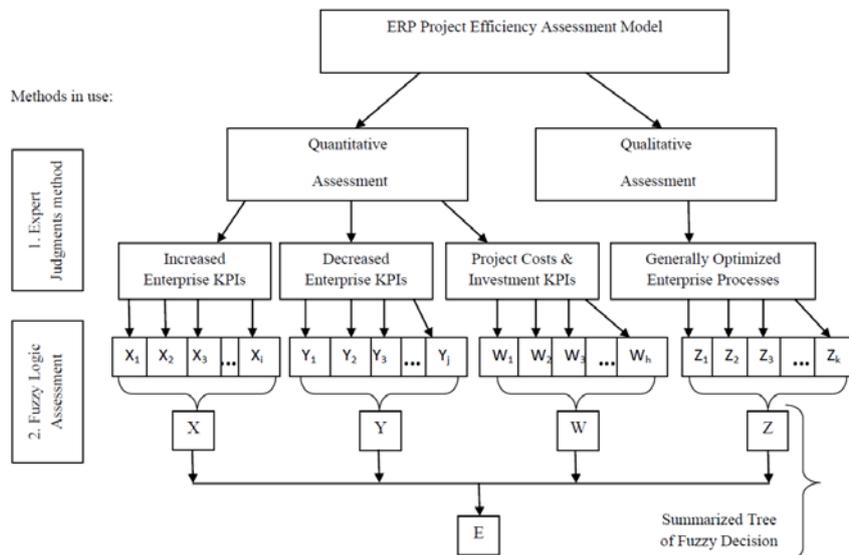


Fig. 1. Combined Model of ERP Projects Efficiency Assessment

Developing a questionnaire for determining the Fuzzy (if-then) Rules with respect to the three optimal KPIs values obtained from Experts as inputs and ERP project efficient value as output. Validity and reliability of the questionnaire was confirmed and they were distributed among the ERP practitioners. Fuzzy rules as the basis for determination of the conditions of the company KPIs have been formed and entered into Fuzzy system through MATLAB software.

Currently there are variety of software tools and system for applying the fuzzy logic calculations available on the market (CubiCalc 2.0 RTC, CubiQuick, FIDE, Flex Tool, FuziCalc, FuzzyTECH, JFS, MATLAB - Fuzzy Logic Toolbox, RuleMaker etc). Each of these products has its own strengths and weaknesses, however as software platform for our research it was decided to go with MATLAB Fuzzy Logic

Toolbox (FLT) as most appropriate tool in particular because of the integrated nature of the MATLAB environment that also provides functions, applications, and a simulative block for analyzing, designing, and simulating systems based on fuzzy logic, widely used not only in academic and research institutions but by industrial enterprises as well.

To determine the value of each factor, different questionnaire was prepared to collect related information for the KPIs values. Also validity and reliability of mentioned questionnaire was confirmed and it was distributed among managers and experts in that domain. Calculated means obtained from questionnaire, results has been inputted to the Fuzzy System (see fig. 2). Final results were analyzed and the efficiency of the implemented ERP project was determined by this software.

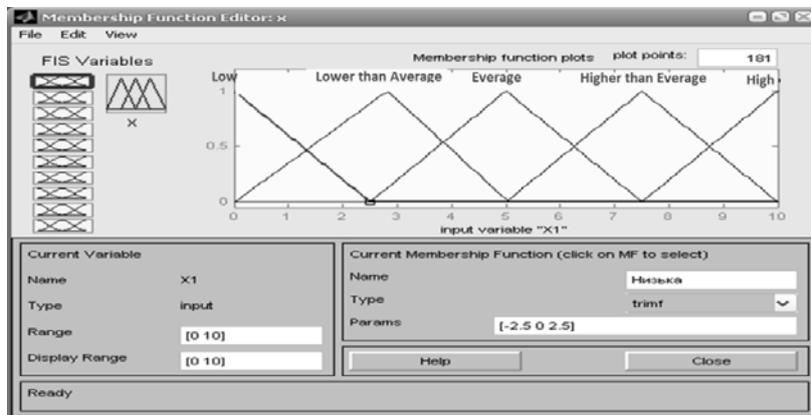


Fig. 2. Identifying the membership functions for each input and output variable

To design a Fuzzy system by MATLAB following: X, Y, Z, W are lingual values that also represents an organizational major KPIs groups and E is the lingual value of summarized project Efficiency (see fig. 3 and fig. 4).

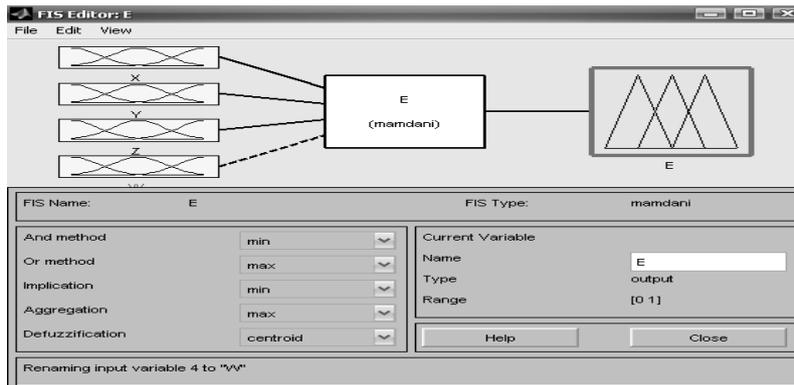


Fig. 3. The Fuzzy system of assessing the efficiency of the ERP implementation in MATLAB software

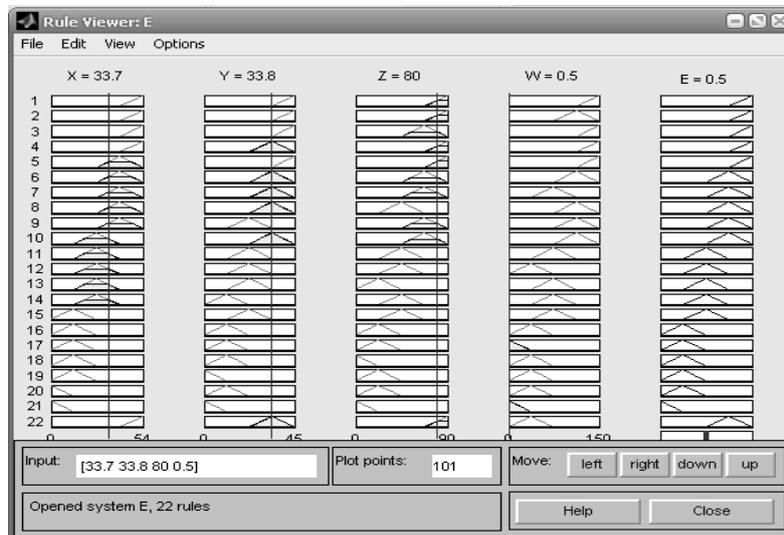


Fig. 4. Drawing a conclusion in Fuzzy system

4 Conclusions

Proposed approach and model of determining the efficiency level of the implemented ERP-project with fuzzy logic can be used by variety of ERP practitioners, project managers, top management personnel of Enterprises that implements ERP-system for advanced analysis of the actual project results. Fuzzy rules should be further validated and formed by consulting with ERP practitioners and their results will be entered into knowledge base of Fuzzy system. Methods of fuzzy logic computing combined with provided expert judgment aimed optimize the speed of the decision-making on project efficiency level and simultaneously provide more accurate assessment abilities.

References

1. Finney. S., Corbett. M.: ERP Implementation: a Compilation and Analysis of Critical Success Factors. *Business Process Management Journal*, 13(3), 329–347 (2007).
2. Allen, D., Ken, T., Havenhand, M.: ERP. Critical Success Factors: an Exploration of the Contextual Factors in Public Sector Institution. In: *Proc. 35th Hawaii International Conference on System Sciences*, pp. 244–247 (2002)
3. O'Leary, D.: *ERP Systems: Modern Planning and Enterprise Resource Management. Select, Implement, Utilize*. Vershina, Moscow (2004)
4. Savavko, M.: *IS Fuzzy Expert*. Publishing House of I. Franko Lviv National University, Lviv (2007).
5. Nozdrina, L.: Applying of Fuzzy Logic Modeling for the Assessment of ERP Projects Efficiency. In: *Proc. 5th Int. Sci. Conf. Project Management: Status and Opportunities*, pp. 1–2, NUS, Nikolaev (2009)

6. Gartner: Information Technology Research and Advisory Agency. <http://www.gartner.com/technology/home.jsp>
7. Panorama Consulting. Consulting Firm with Expertise in ERP Software, <http://panorama-consulting.com>
8. IDC. Global Provider of Market Intelligence, Advisory Services, and Events for the Information Technology, <http://www.idc.com/home.jsp?t=1365517508962#.UWQk-5NTCjg>
9. Austin, R. D.: Measuring and Managing Performance in Organizations. Dorset House Publishing (1996)

Appendix A. Example of ERP User Satisfaction Survey

New ERP-system user satisfaction survey

Dear user ERP-system! This survey aims to study and evaluate employees opinion on the quality characteristics of the new ERP system, and the effectiveness of project implementation in the enterprise as a whole. The data collected will be used to study and model building performance evaluation ERP-project methods of fuzzy logic (in the environment of MATLAB) within Research "Development of methods for managing ERP-projects in the trade." In the expert group of the survey include: managers and consultants ERP-projects, IT department heads, managers, independent experts on the use of fuzzy logic models and scientists). Answer the following questions.

* Required field

Questionnaire

1. Are you satisfied generally introduced official ERP-system *

1 2 3 4 5
 Very dissatisfied (a) Very satisfied (a)

Is there compared to the previous system (s), the new ERP is better? *

- Much better
- Slightly better
- The difference is not felt
- A little worse
- Much worse

2. Please please, how much you agree or disagree with the following statements *

	Yes, I agree (a)	We can say that he agrees (a)	More than agree (a) than disagree (s)	We can say that disagree (s)	I do not agree (s)
The system is generally higher quality and more efficient than the previous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
High level of comfort in working with documents	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A much simpler and more intuitive interface	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Simply and quickly find the information needed by the system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Use, and the process of the new system leaves a positive impression convenient and intuitive interface allows you to quickly open and view the desired report, table, directory, document, etc.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

It appeared in the new system funtsional, and tools that help simplify the execution of your daily work and routine tasks? *

- Yes
- No

Do you agree with the statement that it is possible in the new system some processes and workflow seem more complicated, but also become accessible and more useful for information, and generally operate more effectively become the new ERP-system? *

1 2 3 4 5
 Yes, I agree No, I do not agree

Your Comments

Appendix B. Example of ERP Expert Judgment Questionnaire

Questionnaire. Evaluation of the effectiveness of ERP-projects

Dear expert! This expert survey aims to study and evaluate opinions of experts on ERP systems and projects for their implementation. The collected data will be used to conduct research and build models evaluating the effectiveness of ERP-project methods of fuzzy logic (in the environment of MATLAB) within Research "Development of methods for managing ERP-projects in unsteady economic systems." In the expert group of the survey include: managers and consultants ERP-projects, IT department heads, managers, independent experts on the use of fuzzy logic models and scientists). Answer the questionnaire. We guarantee that your information will be used only for research purposes.
* Required field

1. Please fill in the following personal and contact details: *

Name and Surname:

Title:

Degree:

Contact E-mail address:

2. According to independent international consulting companies, news agencies and collected statistics presented in analytical surveys were selected X, Y, and Z groups, and relevant performance indicators ERP-projects (see Appendix A). Along with the name parameter specified level of performance achieved after the implementation of ERP, in% (based on analytical reviews.) Do you believe that the introduction of ERP-systems on the Ukrainian commercial enterprise segment of small and medium-sized businesses will have specific features that will change the list of indicators in each group?

- Yes
 No

2.1. A few wholesalers of the city, based on key indicators presented in Appendix A, when assessing the effectiveness of ERP-systems have been identified a number of additional indicators relevant to the activities of domestic commercial enterprise. Do you agree that when assessing the effectiveness of ERP-systems according to the commercial enterprise of the list should be supplemented these indicators?

Select the desired parameters, or specify your own version, which indicators should be added, replaced, or removed from the list)

- Turnover (sales)
 Number of new customers
 Number of orders
 AVG orders
 Number of customer orders
 The average order amount
 Turnover by customer
 Number of invoices per month for commodity
 Other:

3. Fill in the fields with indicators, selecting the group, which include the corresponding figure (X, Y, Z or W), and if you have experience and practical knowledge of ERP-systems to provide commercial enterprises - in the likely values of the effective proceedings (eg turnover (sales, thous.) - Group A / 10%).

Turnover of sale, thous. (Group / probable value for the effective implementation of,%)

Number of new clients (group / probable value for the effective implementation of,%)

Number of Orders (group / probable value for the effective implementation of,%)

Mathematical Model of Banking Firm as Tool for Analysis, Management and Learning

Victor Selyutin^{1,2} and Margarita Rudenko¹

¹ Research Institute of Mechanics and Applied Mathematics,

vvs1812@gmail.com, ritusik@mail.ru

² Economic Faculty of Southern Federal University
220/1, av. Stachki, 344090, Rostov-on-Don, Russia

Abstract. An essential concern for banking firms is the problem of assets and liabilities managing (ALM). Over last years a lot of model tools were offered for solving this problem. We offer the novel approach to ALM based on transport equations for loan and deposit dynamics. Given the bank's initial state, and various deposit inflow scenarios the model allows provide simulations including stress-testing, and can be used for assessment of liquidity risk, for examine loan issue decisions to choose reasonable solution, and in the learning purposes.

Keywords. Asset- liability management, Differential equations, Liquidity risk, Duration

Key terms. Banking, Mathematical Modelling, Decision making

1 Introduction

A banking firm is rather a complex system within the context of management problem. It is caused by a considerable number of financial flows and the funds, having a various origin and differing by dynamic and probabilistic characteristics, and at the same time forming the unified system. Stable functioning of the system is provided due to hierarchy, external (prudential supervision) and internal regulators and restrictions, and feedbacks.

Among the mathematical models of banking firms it is possible to separate two basic groups. There are models of optimization of assets portfolio (static, single and multi-period) using linear and dynamic programming mainly [1-2], and models of assets and liability management (ALM), using methodology and the technique of the stochastic differential equations [3-5].

One of the problem solved by models ALM is management of various risks (especially credit risk and interest-rate risk), including the problem of default probability decrease.

In connection with computer engineering development, from the middle of 70th years of the last century the computer models of banks focused on problems of planning and decision-making support systems began to appear. However such projects had no further development [6-8].

Then we will turn our attention to one of the possible approaches to bank modelling as a dynamic system, which can be called hybrid. The main tasks which the model developed must solve are the analysis and management of liquidity and stress-testing of a bank. In addition, it can be used for optimization of assets profile.

Aggregation of elements of balance sheet can be varied according to the objectives of modelling and principles developing of state variables vector. We will use the following simplified schematic (Tab. 1).

Fixed assets of bank we will ignore, taking into account only financial flows. Obviously, balance sheet equation takes place:

$A = S + B + Q + X = Y + C + M = L,$	(1)
--------------------------------------	-----

where equity (capital) of a bank *C* is a balancing variable.

For detailed modelling of credit risks, loans issued can be divided by categories of the debtors having various reliabilities. Division of deposits on time and demand is necessary for calculation of instant liquidity. It is ignored in considered below version of the model for simplicity.

Table 1. The aggregated balance sheet of commercial bank.

Assets (A)		Liabilities (L)	
Loans issued (X):	Business	Debt (Y)	Time deposits
	Private customers (buyer`s credits, mortgage etc.)		On-demand deposits and current accounts
	Other banks	Inter-bank credits (M)	
Securities	Shares (Q)	Equity, including retained profit of last periods (C)	
	Bonds (B)		
Reserves (S)	Cash	Equity, including retained profit of last periods (C)	
	Rest fund, loan loss reserves etc.		

Formally it is possible to mark three groups of operations in the balance-sheet table:

- Reallocation of assets between separate items
- Reallocation of liabilities between separate items
- Identical change of assets and liabilities at one period

Though the bank opens a position in liabilities with grant of a loan (opening of a credit line) at one time, from the formal point of view this operation is resolved into reallocation of asset`s items.

Similarly, if the deposit remains unclaimed in maturity date it either is prolonged, or is transferred in demand deposits (with no interest accruing or with the minimum percentage) according to contact conditions. Actually, in this case there is a reallocation of liability's items.

At last, when interest on loans (or other types of income or expenses) are received (or repaid), at one time it is changed both assets, and liabilities, own capital of bank increases or decreases.

2 Model with Certain Terms of Loans and Attracted Funds

The main difficulty in modelling of assets and liabilities dynamics is concerned with necessity taking into account terms of loans and deposits. Due to these the state variables must depend on two parameters - current time (t) and current "age" (τ) or the remained term to maturity (T-τ). That is why dynamics of the issued loans can be described by following transport equation:

$$\frac{\partial x}{\partial t} + \frac{\partial x}{\partial \tau} = u(t, \tau) \tag{2}$$

In addition $X(t) = \int_0^T x(t, \tau) d\tau$ - total amount loans issued,

$X^*(t) = \int_0^T x(t, \tau) e^{-\delta\tau} d\tau$ - present value of loans, T – term of loans.

Movement of time deposits is described similarly:

$$\frac{\partial y}{\partial t} + \frac{\partial y}{\partial \tau} = v(t, \tau) \tag{3}$$

$Y(t) = \int_0^T y(t, \tau) d\tau$ - total amount of time deposits,

$Y^*(t) = \int_0^T y(t, \tau) e^{-\delta\tau} d\tau$ - present value of time deposits, T – term of deposits.

Variables $u(t, \tau)$ and $v(t, \tau)$ denote the flows of issued loans (temporary outflow of financial resources of bank) and deposits (temporary inflow) distributed by time taking into account amortization (interest payment or installment credits). Accordingly, total inputs of loans $U(t)$ and deposits $V(t)$ (or its present values $U^*(t)$ and $V^*(t)$) is described as:

$$U(t) = \int_0^T u(t, \tau) d\tau, U^*(t) = \int_0^T u(t, \tau) e^{-\delta\tau} d\tau$$

$$V(t) = \int_0^T v(t, \tau) d\tau, V^*(t) = \int_0^T v(t, \tau) e^{-\delta\tau} d\tau$$

Solution of the equations (2-3) can be represented in the closed form:

$$x(t, \tau) = \int_0^t u(\xi, \tau - t + \xi) d\xi + \varphi(\tau - t)$$

$$y(t, \tau) = \int_0^t v(\xi, \tau - t + \xi) d\xi + \psi(\tau - t)$$

where $\varphi(\tau)$ and $\psi(\tau)$ - initial distributions of loans and deposits by "age", or may be obtained by use corresponding equations with finite differences.

Dynamics of reserves (S) and equity (C) is described by the equations including stochastic members which consider random nature of change in value of shares and possible loans losses:

$$dS = [U(t) - V(t) + \rho_X X - \rho_Y Y + \rho_B B - \rho_M M - Z(t)]dt + \mu Q dt + \sigma Q dW_t - x_t dJ_t$$

$$dC = [\rho_X X - \rho_Y Y + \rho_B B - \rho_M M - Z(t)]dt + \mu Q dt + \sigma Q dW_t - x_T dJ_t$$

where dW_t – increment of Wiener stochastic process, dJ_t – increment of compound Poisson process with exponential distributed size of jumps (loan losses), $Z(t)$ – operation expenses and payment for dividends; $x_T(t)$ – repayment of a loans in maturity date, $\rho_X, \rho_Y, \rho_B, \rho_M$ – accordingly interest on loans, deposits, bonds income, cost of credits; μ - average portfolio return of trading securities, σ - volatility of securities portfolio.

Investments in liquid assets - shares $Q(t)$ and bonds $B(t)$ can be considered as some parameters of management and to be calculated, proceeding from structure of assets chosen or planned by bank taking into account loan demand. Similarly, the volume of received loans $M(t)$ can be select depending on bank's requirement in financial resources.

It is necessary to add the equations of dynamics of duration to the equations of movement of assets and liabilities to model liquidity risk taking into account change of interest rates

If r – the annual interest rate, so in this case Macaulay duration for an asset $x(t, \tau)$ is defined by expression:

$$D_x(t) = T - \frac{1}{X^*(t)} \cdot \int_0^T \tau \cdot x(t, \tau) e^{-\delta\tau} d\tau,$$

where $\delta = \ln(1+r)$.

Similarly duration of another financial flows $y(t, \tau)$, $u(t, \tau)$, $v(t, \tau)$ are calculated. It is possible to show that dynamics of duration is described by any of presented below the equations which is chosen according to liquidity research tasks.

$$\frac{dD_x}{dt} = \left[D_u(t) \frac{U^*(t)}{X^*(t)} - 1 \right] - \lambda(t) D_x - \delta D_x$$

$$\frac{dD_x}{dt} = (D_u(t) - D_x) \frac{U^*(t)}{X^*(t)} - \left[1 - D_x \frac{x_T(t)}{X^*(t)} \right] - \delta D_x$$

$$\frac{dD_x}{dt} = \lambda(t)[D_u(t) - D_x] - \left[1 - D_u(t) \frac{x_T(t)}{X^*(t)} \right] - \delta D_x$$

where $\left(\frac{dX^*}{dt} \right) \frac{1}{X^*} \equiv \lambda(t)$

Model (2-3) has been transformed to system of difference equations and realized as computer program [9]. The user independently chooses one of two operating modes of the program: calculation in case of predefined planning horizon, or calculation with possible correction of parameters, setting physical speed of calculation.

The program is interactive as the user can change values of some key parameters in the process of calculation, without interrupting its work. As key parameters are chosen: a fraction of cash invested in various kinds of assets, revenues (interest rates), a duration of demand deposits, credit demand, inflow of deposits, crediting scenarios (distribution of loans by time).

Dynamics of inflows and outflows of cashes; diagram of change of durations of assets and liabilities; distributions of loans and deposits, and also input flow by time are displayed on the screen of computer.

A stress-testing is provided in the program. The user can choose the period of stress-testing and such stresses-scenarios as decrease in inflow of deposits, decrease in duration of deposits (the scenario of outflow of deposits); decrease in accessible volume of attracted funds on the interbank market.

3 Model with Fixed Terms of Lending and Borrowing

Model (2-3) presented above is rather difficult in numerical realization and does not allow to consider some important facts, for example, dependence on interest rates from different terms of lending or borrowing. Therefore we will consider simplified modification of previous model under supposing that terms of loans and deposits are fixed.

It is possible to fix the most typical terms according to the classification used in the bank reporting, in spite of the fact that terms of loans (or deposits) can be arbitrary. Both loans and time deposits are structured by terms as follows: till 30 days, from 31 till 90 days, from 91 till 180 days, from 181 days till 1 year, from 1 year till 3 years, over 3 years.

Accordingly, it is possible to establish several typical periods T_k for each of them time transactions are described by the partial differential equation of the first order

$$\frac{\partial x}{\partial t} + \frac{\partial x}{\partial \tau} = a(\tau, x) \tag{4}$$

with a boundary condition $x(t, 0) = u(t)$ and the initial condition $x(0, \tau) = \varphi(\tau)$. Initial and boundary condition should be consistent, that is $u(0) = \varphi(0)$.

Here t - current time, $0 \leq t < \infty$, τ - elapsed time since the moment of settlement of transaction ("age" of an loan or deposit), $0 < \tau \leq T$, $a(\tau, x)$ - value of "amortization" of an loan or deposit (inflation, installment credit etc.).

Similarly (2-3), the variable $x(t, \tau)$ is the allocated variable characterizing some credit tools, accounted in assets or in liabilities (loans for limited period, time deposits, interbank lending or borrowing, coupon bonds or other assets and liabilities with the fixed term of repayment).

Further it will be assumed that

$$a(\tau, x) = -\varepsilon x \tag{5}$$

i.e. repayment of credits occurs proportionally to their volume with coefficient ε , which is not dependent on age. It can be used and other schemes (when credit repayment begins not at once and (or) occurs in advance established equal shares.

It is easy to verify that the solution of the equation (4) looks like a travelling wave

$$x(t, \tau) = u(t - \tau) \exp(-\varepsilon \tau) \tag{6}$$

For consistency an initial and boundary conditions at $t < \tau \leq T$ it is necessary to predetermine $u(t)$ on an interval $t \in [-T, 0)$.

From (4) - (6) follows

$$x(0, \tau) = \varphi(\tau) = u(-\tau) \exp(-\varepsilon \tau) \tag{7}$$

and after replacement τ for $-t$,

$$u(t) = \varphi(-t) \exp(\varepsilon t) \text{ under } -T \leq t < 0 \tag{8}$$

The total value of the considered loan (or deposit) are obtained by integration on age

$$X(t) = \int_0^T x(t, \tau) d\tau \tag{9}$$

Substituting (6) in (9), we have

$$X(t) = \int_0^T u(t - \tau) \exp(-\varepsilon \tau) d\tau \tag{10}$$

Integrating (4), we obtain the ordinal differential equation

$$\frac{dX}{dt} = u(t) - \varepsilon X - x(t, T) = u(t) - \varepsilon X - u(t - T) \exp(-\varepsilon T) \tag{11}$$

As assets with different terms of repayment are in portfolio of assets or liabilities, so it is possible to replace scalar variable $X(t)$ in (11) with vector. Vector's components are financial tools with different terms of repayment

$$\frac{dX_k}{dt} = u_k(t) - \varepsilon_k X_k - x_k(t, T_k) = u_k(t) - \varepsilon_k X_k - u_k(t - T_k) \exp(-\varepsilon_k T_k) \tag{12}$$

For simplicity further we will suppose $T_k = k$, where k - the term, expressed in months.

Time tools (issued loans, bonds, interbank credits, time deposits) from the mathematical point of view are similar, that is why we will consider them in the context of one and only construction, giving the general designation: X_k - to time tools in assets and Y_k - in liabilities. Then the previous model can be presented as:

$$\frac{dX_k}{dt} = u_k(t) - \varepsilon_k X_k - x_k(t, k) = u_k(t) - \varepsilon_k X_k - u_k(t - k) \exp(-\varepsilon_k k) \tag{13}$$

$$dS_t = \mu S_t dt + \sigma S_t dW_t + f(t) dt \tag{14}$$

$$\frac{dQ}{dt} = \sum_k \frac{dY_k}{dt} - \sum_k \frac{dX_k}{dt} + \frac{dZ}{dt} + \sum_k \rho_k X_k - \sum_k \eta_k Y_k - g(t) - f(t) \quad (15)$$

$$\frac{dY_k}{dt} = v_k(t) - \varepsilon_k Y_k - y_k(t, k) = v_k(t) - \varepsilon_k Y_k - v_k(t - k) \exp(-\varepsilon_k k) \quad (16)$$

$$\frac{dZ}{dt} = w(t) - \frac{Z}{D_z} \quad (17)$$

where $w(t)$ - inflow of on-demand deposits, $v_k(t)$ - inflow of time deposits and borrowed funds; $f(t)$ - purchase (+) or sale (-) trading securities (t/s); $g(t)$ - operation costs on carrying out of activities of bank; μ - securities portfolio return; σ - volatility of securities portfolio; W_t - Wiener stochastic process; η_k - interest on the time deposits and borrowed funds; ρ_k - interest on issued loans; D_z - duration (characteristic turn-over time) on-demand deposits.

It is easily to obtain the equation of dynamics of equity by differentiation of balance equality and corresponding substitutions (13) - (17). As follows,

$$\frac{dC}{dt} = \sum_k \rho_k X_k - \sum_k \eta_k Y_k + \frac{dS_t}{dt} - f(t) - g(t) \quad (18)$$

For simplicity it is supposed complete withdrawal of deposits after term in this version of model. However it is easy to take into account possibility of prolongation of the deposit or its transfer in category on-demand deposits. It is considered that dividends are not paid.

Besides, credit risks (default risk, or a delay of payments) are not considered, that also it is possible to take into account by entering of corresponding adjustments. It is considered that interests on the attracted funds and the received credits are paid according to accrual. However it is easy to set and other scheme in which interests are accumulated on depositary accounts and are paid after term of deposit.

Let $\alpha_k = X_k / X$ and $\beta_k = Y_k / Y$ - structure of time loans and deposits.

Besides, for simplicity we will assume that there are no investments in trading securities. Then dynamics of the capitals are described by the equation:

$$\frac{dC}{dt} = X \sum_k \rho_k \alpha_k - Y \sum_k \eta_k \beta_k - g(t), \quad (19)$$

It is giving evident representation about sensitivity of dynamics of capital to changes of main parameters of assets and liabilities.

Main objective of shareholders and bank management is the increase in capital:

$$\frac{dC}{dt} \rightarrow \max \quad (20)$$

subject to restrictions on financial resources and risks (credit and market, loss of liquidity, bankruptcy).

4 Conclusions

The approach to mathematical modelling of cash flow moving in asset and liability accounts of the commercial bank based on the partial differential equations is novel and has no analogues in the literature. At the same time, the given approach is quite logic as reflects process of change of actives simultaneously in time and on "age". Depending on particular theoretical or practical problems the given approach can be realized in the various modifications, two of which are presented in the article.

As the preliminary testing has shown, the computer program created by use model (2-3) allows provide various simulations, including stress-testing, and can be used in the educational purposes to provide the best understanding of the dynamic processes taking place in banking firm.

It is necessary the further development of the offered modelling approach such as improvement of program tool and also, as required, model detailed elaboration to use these models as part of decision support system for asset and liability management in commercial bank. The modified model (13-18) has been proposed for these goals.

References

1. Chi, G., Dong, H., Sun, X.: Decision Making Model of Bank's Assets Portfolio Based on Multi-period Dynamic Optimization. *Systems Engineering – Theory & Practice*, 27(2), 1–16 (2007)
2. Kruger, M.: A Goal Programming Approach to Strategic Bank Balance Sheet Management. *Banking, Financial Services, and Insurance*. In: Proc. SAS Global Forum, Paper 024–2011 (2011)
3. Kosmidou, K., Zopounidis, C.: Asset Liability Management Techniques. *Handbook of Financial Engineering*, pp. 281–300, Springer Science+Business Media, LLC (2008)
4. Mukuddem-Petersen, J., Petersen, M.A.: Bank Management via Stochastic Optimal Control. *Automatica* 42, 1395–1406 (2006)
5. Mulvey, J.M., Shetty, B.: Financial Planning via Multi-stage Stochastic Optimization. *Computers & Operations Research* 31, 1–20 (2004)
6. Solyankin, A.A.: Computerization of the Financial Analysis and Forecasting in Bank. *Fin-StatInform*, Moscow (1998) (in Russian)
7. Robinson, R.S.: BANKMOD: an Interactive Simulation Aid for Bank Financial Planning. *J. Bank Res.* 4(3), 212–224 (1973)
8. Moynihan, G.P., Purushothaman, P., McLeod, R.W., Nichols, W.G.: DSSALM: a Decision Support System for Asset and Liability Management. *Decision Support Syst.* 33(1), 23–38 (2002)
9. Alekseev, I.V., Selyutin, V.V.: Interactive Computer Model of Bank's Asset and Liability Dynamics. *Terra Economicus* 9(4), Part 2, 42–47 (2011) (in Russian)

**2.2 1st International Workshop on Methods
and Resources of Distance Learning
(MRDL 2013)**

Foreword

1st International Workshop on Methods and Resources for Distance Learning (MRDL) has taken place on 19-22 of June 2012, in Kherson, Ukraine in conjunction with 9-th International Conference on ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer (ICTERI 2013).

Distance learning is an important application field that intensively uses information and communication technologies in education. MRDL workshop will bring together reports dealing with the problems of resource maintenance, pedagogic and didactic methods of use of distance learning technologies.

The scope of the MRDL workshop includes the following topics:

- **Virtual laboratories:** Mathematical and informational models and educational tools for virtual labs, covering in particular the distance learning courses in physics, chemistry, biology and other disciplines.
- **Design and development of electronic learning tools and resources:** Design, development and use of electronic learning resources. Compatibility and integration of electronic resources in distance learning systems.
- **Computer-aided learning systems:** Design, development and use of computer-aided learning systems.
- **Pedagogical innovations in distance learning:** Experience in developing and implementation of new pedagogical methods of distance learning using ICT. Open distance courses and tutors training in author's course content.
- **Monitoring of learning quality:** Methods and tools for the estimation of quality of knowledge in distance learning systems, testing, rating systems, feedback.
- **Quality of electronic learning resources:** Standards for electronic resources for distance learning. Modeling of and experience in using quality management systems for electronic learning resources.
- **Information of teaching and educational institutions:** Modeling and experience in the use of management systems in information processing and other management processes at educational institutions. Experience in the use of distance learning technologies in a traditional educational process.

A blind peer-review process by at least two reviewers with expertise in the area has been carried out. As a result, 22 submissions have been accepted as reports, 3 of them are presented in this edition. We would like to thank the authors for their submissions and our Program Committee members for their reviews.

June 2013

Vladimir Kukharenko
Yulia Zaporozhchenko
Hennadiy Kravtsov

What Should be E-Learning Course for Smart Education

Natalia V. Morze¹ and Olena G. Glazunova²

¹ Borys Grinchenko Kyiv State University

n.morze@kmpu.edu.ua

² National University of Life and Environmental Sciences of Ukraine

e_glazunova@yahoo.com

Abstract. The article deals with problems of creation and use of e-learning course for smart education. Structural features, the ratio of form and content of the smart course elements and its properties: individual learning paths, content personification, the use of training elements with links to public information resources, interactive training elements, multimedia, communication and cooperation elements are substantiated.

Keywords. Smart education, e-learning course, informal learning, individual learning path, services of social networks, Content Learning Management Systems

Key terms. KnowledgeEvolution, KnowledgeManagementMethodology, Didactics, KnowledgeManagementProcess, ICTInfrastructure

1 Introduction

Modern information society is gradually transformed into Smart Society, as noted by sociologists, philosophers, specialists in IT sector, educational specialists, etc. This concept implies the new quality of society, in which a set of technological means, services and Internet used by trained people, leads to qualitative changes in the interaction of subjects that allow receive new effects – social, economic and other benefits for a better life [1].

During Smart Society formation the paradigm of education and educational technology is naturally changing. The tasks of training of the new format specialist, successful and competent to work in the Smart Society rely on the new universities – Smart Universities where the integration of technological innovations and the Internet can provide a new quality of the educational and scientific processes, the results of training, scientific, innovation, educational, social and other activities.

The conceptual basis of the Smart University is a large number of different scientific sources, and information and educational materials, multimedia resources (audio,

graphics, video) that can be easily and quickly designed, assembled to a certain set, adjusted individually for each student, his/her need and peculiarity of educational activity and the level of educational achievements.

2 Problem

It is obvious that in conditions of development of Smart Society the educational paradigm will also change. Smart Universities will perform new functions. Accordingly, the requirements for e-learning courses that ensure students' needs in educational resources will change. Our mission is to substantiate theoretically the properties of such e-learning course, its structure and components, and to test the effectiveness of its use experimentally as well.

3 The Presentation of the Main Research and Explanation of Scientific Results

3.1 Characteristics of the Smart University

5 key characteristics of the Smart University can be distinguished: social orientation, mobility, accessibility, technological effectiveness and openness [2].

Social orientation consist in the personalization of education, building of the individual education cards (Smart-card), organization of the efficient communication and collaboration in education, cooperation, application of design and game techniques, communication via social networks services, etc.

The second, equally important, feature of the Smart University is *mobility*. Mobility should be understood not only in the narrow interpretation - as an access to the educational content through mobile devices and their use for scientific researches, payment transactions, implementation of feedback with the teacher or the representatives from the dean office or departments, etc [3]. Mobility is important as an access of each student and teacher to the educational services from any place and at any time.

Accessibility as feature of the Smart University is characterized by a single point of entry to e-learning and scientific databases, media library, information kiosks, online resources and access control systems to them, etc.

Technological effectiveness provides a viability of the Smart University IT infrastructure by the means of cloud-technologies, innovative technologies of virtualization, open interfaces, based on the principles of simplicity, modularity, scalability, etc.

Openness in the system of the Smart University foresees availability of the open repositories of educational materials for forming e-learning courses and providing training for students, open access to scientific articles and conducted researches and their results [4].

3.2 Infrastructures of the Smart University

Modern university should have the appropriate infrastructure to support the requirements of the Smart Education. In particular, the activity of e-learning center, multimedia center, scientific laboratories with the relevant open virtual environments and open resources, library, including electronic one with the open access to the resources, multimedia classrooms and computer labs should be based on the use of advanced campus network with Internet access, including one on the basis of wireless technologies, cloud infrastructure, technologies of mobile access to e-learning resources, system of distributed access to the resources. The effective functioning of such a sophisticated infrastructure is impossible without a united center of data processing, from where materials are distributed to the structural units: institutions, faculties and departments, regional branches, dormitories, academic buildings, student centers, etc. (Fig.1).

Effective activity of the Smart University will enable to realize not only the tasks of formal, but nonformal and informal training as well. According to studies [5,6] nonformal and informal learning takes 70% in the total structure of the educational process and only 30% of learning is formal, in other words structured by years and semesters of training, learning plans and programs that is usually provided by the institution.

3.3 Features of the Smart Education

Smart Education sets a number of tasks for teacher on which performance depends the effectiveness of teaching and students motivation to nonformal and informal learning, which is based on the students skill to study independently. To interest the modern student, who has access to a large number of high quality modern electronic materials that can be easily found in the Internet, by conventional text linear (non-multimedia) materials, only presented in electronic format, nowadays is almost impossible, especially in formal training. We should create such resources that will integrate multimedia, text, feedback tools on the basis of specific teacher's individual recommendations and external electronic resources that will meet individual needs and characteristics of the modern student - regular user of the Internet resources and social networks. Therefore the integral components of information and educational environment of the modern university should be: institutional repository of knowledge with full-text electronic educational and scientific resources; educational portal, which provides electronic support of all student's learning activities for each discipline in the form of e-learning courses with individual tasks and distinct and clear evaluation criteria that are implemented with tools and methods of forming assessment; video portal with multimedia resources for teaching and research activities; wiki portal as an environment to provide teamwork and collaboration; online services based on the use of Web 2.0 and Web 3.0 services and technologies, etc.

One of the main tendencies in the development of Smart Education is the openness of learning systems – placing the educational content openly available to students around the world, the development of systems with open code, development of

knowledge-sharing under the scheme "student-student", "teacher-teacher", "students-teacher" and "students-teachers" [7,8]. An important step in the development of the idea of massive open electronic courses was the adoption of the UNESCO declaration on global policy on the issue of open e-courses, which sets the task of developing standards for electronic courses, providing synergy in access to them, conducting educational seminars on the development of courses and their use, collaboration between scientists and teachers, education quality assurance [9].

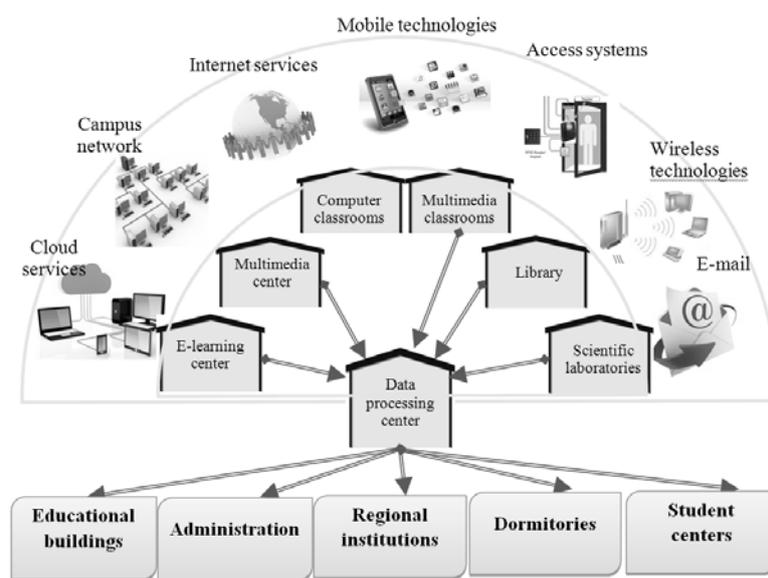


Fig. 1. SMART University infrastructure

3.4 Properties of E-Learning Course for Smart-Education

Some scholars define electronic course as didactic computer environment that contains classified material from the relevant scientific and practical field of knowledge that is combined by a single software shell, in which the following functional components are selected: information and navigation (meaningful connections, annotation and course structure, information, system of references, the searching system), informative (interrelated informative elements of the course – theory, practice, guidelines, additional materials, information resources, including electronic and open), diagnostic (formative assessment tools in the form of clear evaluation criteria for all types of students activity, including self-assessment and mutual assessment, evaluation not only of academic achievements of students, but also evaluation of formation of skills of the 21-st century – to solve problems, work in team, communicate effectively and collaborate, etc., the testing system of current, intermediate and final control) [10]. Electronic course for Smart education should provide flexible learning of the students in an interactive learning environment, which allows him to adapt quickly to the envi-

ronment, to study in any place, at any time on the basis of free access to content all over the world. In our opinion, the electronic course for Smart Education can be represented as a certain scenario or trajectory of educational events how to work with electronic resources in the form of knowledge-map that leads to the achievement of learning effect and has the following properties:

- Flexibility – enabling rapid resources editing and making adjustments in educational trajectory
- Availability of individual learning scenario, in other words, the possibility to draw up an individual educational program for each student from the set of training elements
- Integration of training elements with other open information resources
- Focusing on the learning needs of the student, the personification of content
- Interactivity of learning elements of the course, the maximum use of multimedia technologies (videocasts, animation, video tutorials, screencasts, etc.)
- Feedback between the teacher and the student in the course
- Availability of training elements to ensure effective communication and cooperation of students between themselves and with the teacher, in particular based on the design technology [11]
- Availability of game educational elements
- Providing communication through modern services of social networks [12]

Creation of e-learning courses usually is carried out with the help of Content Learning Management Systems. To create an effective e-learning course for Smart education not only available electronic resources of information and educational environment of the University should be used, but also open external information resources and Web services that will serve as sources of educational and informational materials for electronic course and as means of communication and cooperation (Fig.2).

Information and educational environment of the university should be focused on solving the problem of joint creation and use of academic knowledge for the needs of students and teaching staff of the university. On the one hand, the teacher by himself adds academic resources to the information and educational environment, such as video clips and video tutorials posted on educational video portal and on the other hand, he has the possibility to use available public resources for creating e-learning course. So, to create an electronic course it is sufficient for the teacher to actualize material that is available from other sources, submit it in accordance with the above mentioned properties and criteria of evaluation of its quality, add the necessary training elements of the course according to the adopted structure and develop an individual learning scenario for each student, consider the individual evaluation criteria of educational achievements of students and developed skills of the 21st century.

3.5 Structure of E-Learning Course for Smart-Education

Analysis of papers devoted to the creation and use of e-learning courses [13, 14] led to the conclusion that in the issue of the course structure they should be focused on the modular principle of its construction. When structuring the content of educational

subject by the principle of training modules each module should consist of interconnected theoretical, empirical and practical components of the content, each of which would carry out an independent function. Thus the educational discipline module is an information and didactic unit, in which the approach to structuring the whole into parts is unified. It has a complex structure that includes goal of its integral development, objectives, content and results with the corresponding system of formative assessment.

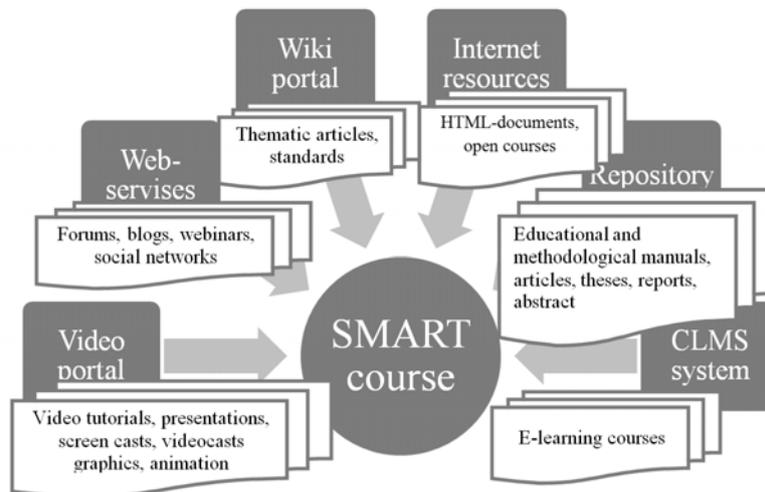


Fig. 2. Sources of electronic course formation for Smart education

Furthermore, the structure of the e-course for Smart education should provide availability of:

- Tools to build individual learning trajectory (prior surveys, questionnaires, tests, formative assessment tools, including check-lists and tables of evaluation criteria, etc.)
- Multimedia presentations of summarizing character, video resources, interactive electronic manuals, external Web resources with multimedia theoretical material
- Links to external public resources including articles, conference proceedings, research materials, etc.
- Discussions on the forums, feedback with teacher, webinars and other Web services
- Intermediate control elements during the lessons and formative evaluation instruments, final control in the form of control tasks and tests, element of reflection

Each element of the training course must meet certain standards and be evaluated using criteria that are accepted at the level of educational institution [15].

Approximate structure of Smart course is shown in the Figure 3.

Example of the course topic, created in the CLMS Moodle environment, presented on training and information portal NUBiP Ukraine (<http://it.nubip.edu.ua/course/view.php?id=21>).

Further we will focus in more detail on the features of the e-learning course structure for Smart education.

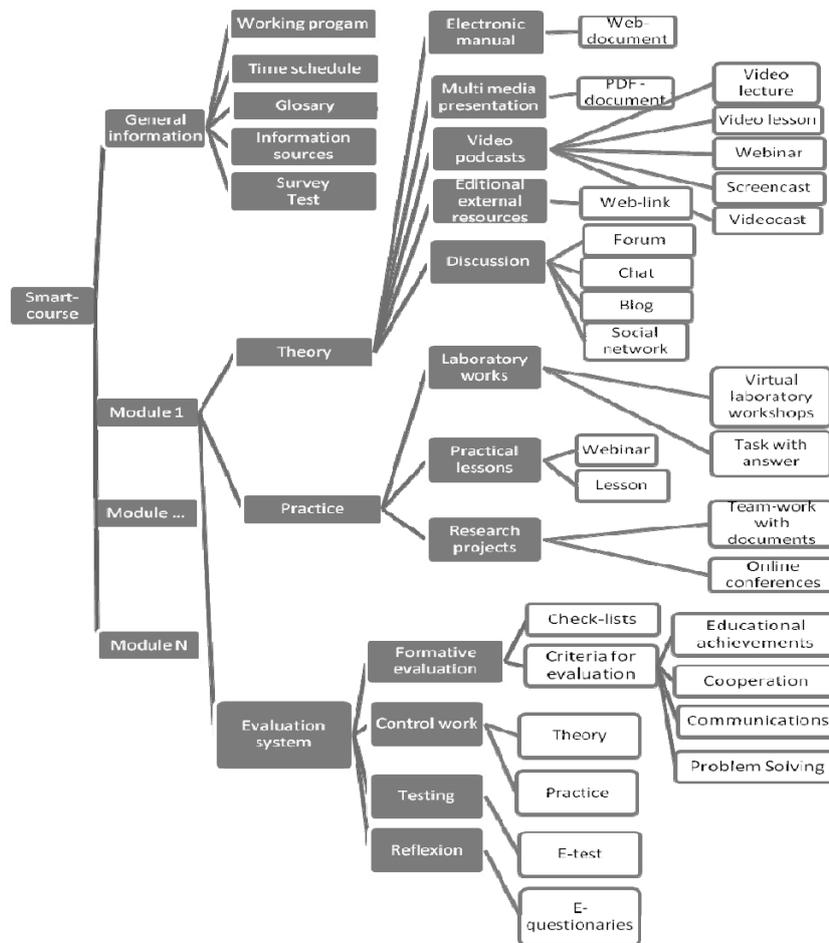


Fig. 3. Structure of the electronic training course for Smart education

3.6 Formation of Individual Learning Trajectory

For the modern student, who has formed basic IT competences, there is a need not only in the access to the resources, but mostly in the navigation knowledge-map, "guidebook" to knowledge, that can be found in information space, as it is important to help student to find quality resources. And this is a complex task for untrained student. Smart education using Smart courses of the new model is the most comfortable and modern teaching model for such cyber-students. To build individual training

trajectory of the student in the electronic course you can use several approaches. One of them lies in the prior survey and testing of the students in terms of competence in the course educational material and the preparation of educational trajectory on the results of such survey and their identified learning needs (Fig. 4). Thus the survey is based on extensive use of formative assessment tools that provides self-assessment and mutual assessment.

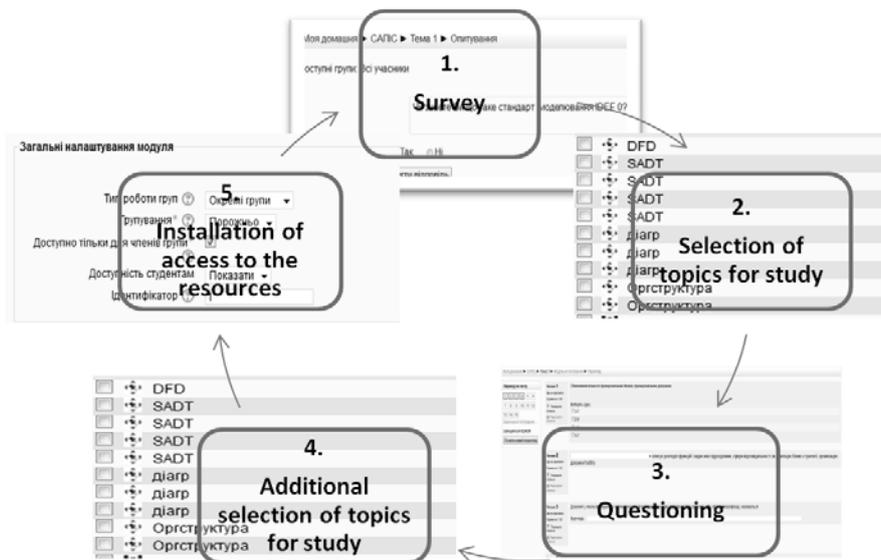


Fig. 4. Stages of the individual trajectory construction

During the experimental study of the introduction of e-learning course of the new sample for the students of the "Computer Sciences" specialty a survey was conducted for assessing their competencies on the subject under the scale: "have a good knowledge", "be partially familiar", "heard something", "not familiar". Then each student was offered a test for competence in the training material, which he/she "has a good knowledge of" and "is partially familiar" with. According to the survey and testing results individual learning trajectory was built for each student or group of students. In other words, sequence of learning elements of the course was chosen, which student should study. Moodle platform that we used to create the course allows to make each training element available to a particular group of students. Therefore each student or group of students receives an individual set of training elements of the course. Training course is adapted for personal characteristics of each student that allow to implement personally oriented approach and to develop an individual training program. At the same time the course itself does not changed, but the methods of presentation, set of tasks for performance and the tools, methods of evaluation and control are changed.

3.7 Presentation Educational Material in the Theoretical Resources of Smart Course

Peculiarity of the new model electronic course is also the diversity of the theoretical learning resources presentation forms. Besides the theoretical material must be delivered by 60-70% in the multimedia interactive form, we also note the necessity to choose the method of material delivery, depending on the level of its teaching. The theoretical material in electronic course can be delivered on the following four levels: *phenomenological, analytical and synthetic, mathematical, axiomatic* [16]. Each level has its peculiarities in the delivery of educational material (Fig. 5).

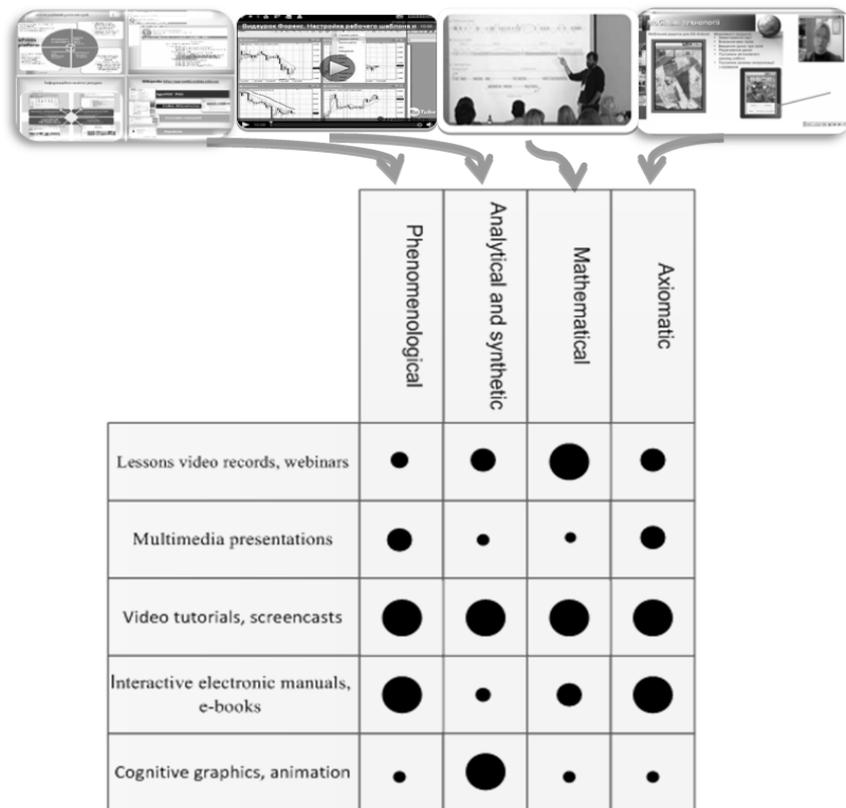


Fig. 5. Ratio of levels and methods of educational material presentation in the theoretical resources

Phenomenological level is characterized by the descriptive way of presenting educational material. Therefore, these materials should be delivered in the form of multimedia presentations, interactive electronic manuals with graphics, multimedia and video elements. Analytic and synthetic level is characterized by the necessity of presenting of the theory of individual phenomena in naturally logical language that cre-

ates background for phenomena and processes forecast on a qualitative level. For this level animation resources with elements of cognitive graphics should be prepared that will be able to demonstrate the nature of the phenomenon and its dynamic changes. Video tutorials with explanation and demonstration of the logic of the processes as well as sound screencasts will be also effective. The mathematical level is characterized by the use of mathematical tools for modeling, theorem proving, examples of solving problems, etc. Therefore, conventional textbooks are not enough to deliver such material. It is necessary to create resources in the form of video lectures, video lessons, and text resources should be reduced to the minimum amount – in the form of handbooks with basic rules, formulas, theorems, etc. Educational material of the axiomatic level can be presented in the form of video tutorials, e-manual and multimedia presentations. Also it is necessary to actively use the links to external resources that cover material from the considered topic. Such resources will add credibility to the course and allow students to familiarize with additional sources of educational materials.

3.8 Presentation Learning Tasks in the Smart Course

Another feature of the e-course for Smart education is the existence of elements for communication and cooperation between the students in the performance of tasks of mastering theoretical material, practical tasks, research projects, etc. Web 2.0 services, online services, social networks provide tools for organizing discussions, collaboration, counseling. These elements are embedded in the course directly through the platform that is used, or by reference to it. While performing tasks students should use modern information and communication technologies effectively. Usually for **mastering the theoretical teaching material** students (Fig.6) are offered tasks on writing essays, composition of the related bibliography, writing summaries, paraphrasing theoretical information of a small amount in the form of "question-answer", compiling a glossary of terms under the certain topic, performing descriptive works, making instructions for the implementation of various operations, plotting grid plans and schedules.

In order that such tasks become interesting for students it is necessary to use Internet resources, and present the result of the performance in the electronic form using modern information technologies.

Tasks **on mastering practical skills** include solving problems, performance of exercises, graphical works, practical works, calculated works, designing, modeling, compilation of practical situations from their own experience and on the basis of practical training, performing analysis of enterprise activity. Students should be offered to solve such tasks using virtual laboratory workshops and specialized software.

Tasks on forming **research activity** include implementation of individual research tasks, writing term papers, graduation works, participation in the educational projects. Such types of tasks involve creative activity of the students, which should be carried out by means of modern computer technologies, teamwork and communication. And one of their peculiarities is the use of formative assessment tools for their assessment that clearly guide the student to achieve learning goals in all types of educational

activity that are presented specifically, clearly and should be achievable for each of them.

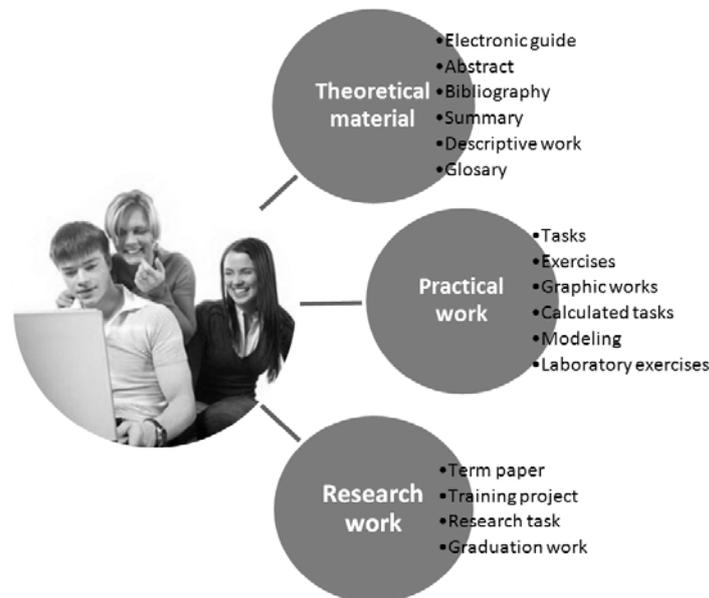


Fig. 6. Types of tasks to master practical skills

Thus, the main features of the students practical work organization using e-course in the Smart education is the availability of tools for collaboration, communication, combination of different information technologies in the performance of tasks. But we should not forget that tasks should have practical significance, and contain detailed information on their implementation, evaluation criteria and support resources. We offer such pattern for the formulation of the task in the e-course (Fig.7)

3.9 Results of Experimental Research

In the course of research conducting we proposed a new model of electronic learning course for students of shortened training period. As a result, after questioning and testing six groups of students were identified who studied in different educational trajectories, successfully completed training course and demonstrated 13% better academic progress compared to the group of students who studied in one training trajectory. At the same time, the participants of the experimental groups performed larger volume of tasks on the depth study and worked extra theoretical study material according to the references to the external information resources. In addition, teachers-participants of the experiment, indicated that the presence of distributed environment of the opened resources in the university allows to create e-learning courses applying much lesser efforts and requires lesser time compared to the case when the

course is created from the beginning. Teachers are able to use ready resources for creating elements of the course – presentations, video recordings, electronic versions of manuals and guidelines, a database of scientific publications, etc.

1. *What to do?* (clear formulation of the task)
2. *In what order?* (algorithm of task performance, progress of work, methodology of the task performance, examples of such task performance)
3. *What materials to use for the task performance?* (links to sources that contain necessary information for task performance).
4. *What tools to use for the task performance?* (standard or customized software, summary and pen, special equipment, etc.).
5. *In what form to submit fulfilled task?* (comparison table, presentation (photo album), booklet, a list of useful resources to the topic, organizational charts, graphs and diagrams, a collection of images or photos, article, cause and effect diagram, scheme of concepts to the theme, timing diagram, test questions, Web site).
6. *How performed task will be evaluated?* Criteria of task evaluation it is appropriate to spell out in tabular form:

Task element	Evaluation criteria	Number of points
Table construction.....	Properly chosen	5
...
Total		20
7. *What is the time limit for task execution?* (term to which performed task should be submitted, and it is necessary to adhere to these deadlines when works test and valuation).

Fig. 7. Pattern for the formulation of the task in the e-course

4 Conclusions

Thus we note that the electronic course that has the properties required in the view of Smart education is an effective tool for nonformal and informal learning, in which most motivated students are interested now for obtaining high-quality knowledge, not only a diploma of higher education. For efficient organization of learning activity in the conditions of Smart education modern university should have distributed information and educational environment that will enable to concentrate open electronic learning resources and to move knowledge into a distributed network, actively use the Web 2.0 services, mobile technologies, management system for learning content for delivering knowledge to the students and the interactive exchange of information data and training materials with them. In the future the development of such approach is possible due to the joint development and use of the open educational content repository by the universities based on the technologies of the Smart education.

References

1. Tikhomirov, N.V.: Global Strategy for the Development of Smart-Society. MESI is on a Smart-University, <http://smartmesi.blogspot.com/2012/03/smart-smart.html> (In Russian).
2. Measuring the Information Society 2012, Committed to connecting the world, http://www.itu.int/dms_pub/itu-d/opb/ind/D-IND-ICTOI-2012-SUM-PDF-R.pdf (In Russian).
3. Traxler, J.: The Learner Experience of Mobiles, Mobility and Connectedness. Evaluation of Learners' Experiences of e-Learning Special Interest Group. <http://www.helenwhitehead.com/elesig/ELESIG%20Mobilities%20ReviewPDF.pdf> (2010)
4. McAuley, A., Stewart, B., Siemens, G., Cormier, D.: The MOOC Model for Digital Practice. http://www.elearnspace.org/Articles/MOOC_Final.pdf
5. Kuharenko V. M.: Formal, Informal, Informalne and Social Studies. In: Modern Educational Technology in Education, pp. 114–124 (2012) (In Russian)
6. Mapping Informal and Formal Learning Strategies to Real Work. <http://performancedesign.wordpress.com/2011/05/04/mapping-informal-and-formal-learning-strategies-to-real-work>
7. Helmer, J.: A Pair of Key Trends for this Year Learning: MOOCs and OA, <http://www.smart-edu.com/moocs-and-oa.html>
8. Open Educational Resources, <http://www.unesco.org/new/en/communication-and-information/access-to-knowledge/open-educational-resources/>
9. Pawlowski, J.M., Hoel, T.: Towards a Global Policy for Open Educational Resources: The Paris OER Declaration and its Implications, White Paper, Version 0.2, Jyväskylä, Finland, (2012)
10. Bezdolny A.V.: Model of E-Learning Course as a Means of Organizing the Self-Training, <http://cyberleninka.ru/article/n/model-elektronnogo-uchebnogo-kursa-kak-sredstva-organizatsii-samostoyatelnoy-podgotovki> (In Russian)
11. Gnedkova O., Lyakutin V.: Methodological Recommendations of Internet-Services Usage in Distance Learning System “Kherson Virtual University”. *Information Technologies in Education*, 10, 183–187 (2011) (In Russian)
12. Ravenscroft, A.: Dialogue and Connectivism: A New Approach to Understanding and Promoting Dialogue-Rich Networked Learning. *International Review of Research in Open and Distance Learning*, 12(3), <http://www.irrodl.org/index.php/irrodl/article/view/934>
13. Osin A.V.: E-Learning Resources in a New Generation of Questions and Answers, <http://www.ed.gov.ru/news/konkurs/5692#g10> (In Russian)
14. Mosher, B.: Five Myths About Informal Learning, <http://www.smart-edu.com/statikorporativnoe-obuchenie/pyat-mifov-o-neformalnom-obuchenii.html>
15. Morze N. V., Glazunova E. G.: Quality Criteria for E-Learning Courses. *Information Technologies in Education*, 4, 63–76 (2009)
16. Deryabina G. I., Losev V. Yu.: Creating E-Learning Courses: Studies. Samara. Univers – groups (2006)

TIO – a Software Toolset for Mobile Learning in MINT Disciplines

Daniel Sitzmann¹, Dietmar P.F. Möller¹, Karsten Becker² and Harald Richter³

¹University of Hamburg, MIN Faculty, FB Informatik, AB TIS, Building F,
Vogt-Kölln-Str. 30, 22527 Hamburg, Germany

{sitzmann, dmoeller}@informatik.uni-hamburg.de

²Hamburg-Harburg University of Technology, Institute of Computer Technology,
Schwarzenbergstr. 95E, 21071 Hamburg, Germany

k.becker@tuhh.de

³Clausthal University of Technology, Arnold-Sommerfeld-Str. 1, 38678 Clausthal, Germany

richter@tu-clausthal.de

Abstract. A web-based tool-set called TIO was created for mobile learning with emphasis on study courses in mathematics, informatics, natural sciences and technology (MINT). Mobile learning is a variant of E-learning which is based on mobile user devices with Internet access. It was tested for numerous software and hardware configurations users may have and proved to be technically working. TIO consists of a modified version of the open-source E-learning system ILIAS and a tool set. The experience with TIO was that mobile learning is useful for MINT subjects provided that numerous end user devices are supported and several text systems as well. We found that mobile learning is especially useful for professionals because they can learn in their free times in a flexible way. Finally, we found that an emotional component should be existent to make mobile learning more lasting.

Keywords. Mobile learning tool, MINT, single-source publication, social media

Key terms. KnowledgeEvolution, KnowledgeManagementMethodology, Didactics, KnowledgeManagementProcess, ICTInfrastructure

1 Introduction

Mobile learning or M-learning is a new form of E-Learning. Mobile learning means that pupils, students, or professionals are learning via mobile devices such as notebooks, E-books, handhelds, PDAs, smartphones, tablet PCs, iPads, iPods or gaming consoles. The term MINT stands for mathematics, informatics, natural sciences and

technology. The combination of both, i.e. the application of M-learning in MINT subjects is not yet found in literature but addressed in this paper. Because there is no software that supports M-learning for MINT subjects, it had to be developed. The scientific questions of this project are: how should a tool-set look like that optimally supports M-learning for MINT subjects, and what are the benefits and disadvantages of teaching MINT subjects by means of M-learning in general. The latter question will be answered in the future because TIO can be used for a subsequent pedagogical evaluation of the effect and the adoption of M-learning in MINT subjects. It is the technical basis and thus a prerequisite for assessing the benefits and disadvantages. The first research question will be answered in the following because tool-set called “Technical Informatics Online” (TIO) is presented which is a software platform for computer-aided distance learning that is web-based and that has an emphasis on MINT disciplines and on mobile learning. TIO is a front-end for authors of teaching material, and at the same time it is a user interface for learners. It provides for both groups spatial and temporal flexibility in creating content and in consuming it.

Basically, TIO is for the editing of teaching material, its distribution to various mobile end-user devices, for managing study courses, and for learning these course materials. TIO supports so-called single source publishing and serves as a social media for its users in order to make learning a deeper and thus longer-lasting experience. Single source publishing allows to create, to maintain, to retrieve and to deliver the very same content for many heterogeneous end user devices while it is stored only one time in one file. In order to achieve this, a TIO-internal XML-based data structure called “xml4tio” was defined that can be converted by TIO tools into various output formats which optimally support the respective end user device. Because these devices are very different in their capabilities, several presentation formats must be generated out of the same source file, depending on the user's preferences and device type.

TIO is based on a modified version of the open-source software ILIAS and of a TIO web application called TIOWA. It provides for numerous E-learning features, including chat rooms, multiple-choice tests, and for the management of the teaching content, the learning courses as a whole and their users.

The rest of the paper is organized as follows: chapter 2 describes the state-of-the-art in M-learning, single source publishing and social. In chapter 3, An overview description of TIO is given that explains its software components and used technologies. In chapter 4, the technical set-up of TIO is explained by means of block diagrams and xml data structures in more detail. Also the made extensions to ILIAS are described briefly here. In chapter 5, a report about field tests of the tool-set is given. The paper ends with a conclusion, an outlook to future work and a reference list.

2 State-of-the-Art

2.1 M-Learning

M-learning is a modification of E-learning for the purpose of distance education and blended learning based on mobile end-user devices. An overview on M-learning can be found in [4], [5], [6] and [7]. Blended learning means that customers must show-up in a classroom for a fraction of about 20% of their time and are not allowed to study completely from remote.

In theory, M-learning has several advantages: First, mobile devices are already widespread. Thus, no or little investment is needed for the user compared to a desktop PC. This is important for many young clients and for customers from developing countries. Second, mobile devices reflect the life style of the generation of 'digital natives' [8]. Thus the potential acceptance and use of M-learning may be higher than for classical learning styles. Third, mobile devices help professionals to consume content on top of their working hours for the purpose of life-long learning because of the access flexibility they get. They can use free time slots while travelling between work and home for learning, or weekends and holidays in a very flexible way. Fourth, Internet access allows quicker distribution of content and for significant lower media costs compared to all other distribution ways that are based on paper. Fifth, if the sensors of the end-user device such as the GPS position, the tilt- and acceleration-meters are engaged then learning can adapt dynamically to the current location and situation of the learner.

In practice, M-learning faces several problems: First, small screen and key sizes hamper its applicability in all cases except of notebooks. As a consequence, a sophisticated layout and formatting of teaching material becomes less important because such formats may not be displayed. Additionally, limited user input must be tolerated because of keyboard restrictions. Second, a rel. slow Internet access and a limited battery life of the user device must be taken into account. Third, heterogeneous hardware and operating systems with no or small hard disks are common. Mobile devices have more options with respect to CPUs, RAM size, displays and operating systems compared to desktop PCs. This makes it very difficult to present and use teaching content equally on all devices.

Numerous E-learning platforms are already in usage. An overview can be found in [9], for example. However, most of them are intended for use on desktop PCs and not for mobile devices. Furthermore, their features for presenting technical content with formulas for visualization of simulation results and for access to remote laboratories is mostly limited. Finally, Internet access via mobile devices is not explicitly supported. TIO addresses these problems and supports text types that occur frequently in MINT subjects.

2.2 Single Source Publishing

Single source publishing means that one internal storage format is used out of which diverse customer outputs can be created, such as html with various cascaded style

sheets and pdf. By single source publishing, a simpler updating process of content is possible that allows to reuse the same teaching material for many devices and in various contexts.

Traditionally, single source publishing is implemented by a 1:1 correlation between chapter and character formats in the source texts and, for example, the generated html code for the user. Additionally, source texts may be augmented with tags and comments that give meta information about the text. From this semantic data, converters can create automatically various output formats, provided that sufficient meta data exists.

TIO uses a combination of both. First, it converts teaching materials that is formatted in a traditional way by chapters, sub chapters and emphasizes such as underline, italic and bold into the common intermediate storage structure xml4tio. Subsequently, it converts xml4tio into the needed user formats. Finally, it uses ILIAS and a self-developed web application called TIOWA to disseminate the requested content in the desired format via mobile Internet.

2.3 Social Media/Web 2.0

Social Media is a generalization of the term web 2.0. It denotes that users are not only consumers of content via download but also producers via upload. It denotes furthermore that users are interacting with each other in the web. Examples of popular social network services are Facebook, twitter, YouTube and Wikipedia. TIO strives to be a social media for distant learning in the MINT disciplines in order to make learning a deeper experience because the distinction between learning and leisure are more flexible then, and because learning gets hereby an emotional component that exists also in a classroom but not in a computer system.

3 Description of TIO

TIO combines E-learning via the Internet with mobile communication. It allows thus for learning from anywhere and at any time. TIO is focused but not limited to mobile user devices. It can be used also with desktop computers. Content creation can be accomplished by authors via the commercial text system “Adobe Frame Maker” [1] if technical manuscripts have to be written. A TIO tool converts these Frame Maker texts in xml4tio. If the editing of mathematical texts is required then a self-developed tool called LearnDSL (Learn Domain Specific Language) can be used for the convenient writing of formulas in a style similar to LaTeX but more user-friendly. LearnDSL converts mathematical texts into xml4tio. Finally, Open Office Writer [2] is supported by TIO for all other use-cases. A macro-based converter transforms Open Office texts again in xml4tio.

TIO is based on the wide-spread ILIAS E-learning platform [3] for which several adaptations and extensions were developed. ILIAS implies PHP [10] as programming language and SCORM [11] as a content format. TIOWA in turn allows to upload teaching material and to convert it from xml4tio into SCORM and html. TIOWA uses

Ajax [12] which provides for the students to interact with TIO as if it were a local application running on his hand-held device. Ajax employs several technologies such as Javascript [13], XML [14], HTML5 [15] and CSS3 [16]. An important part of TIOWA are several converters which are based on XSLT [17]. In principle, TIOWA and ILIAS could even be used independently from each other for the creation, storing and dissemination of teaching material. For example, TIOWA could also be connected to other learn management systems such as Moodle [24].

Aims and Features. Our extensions to ILIAS provide for the following features:

First, authors can create teaching material in that text system that is preferable for them and their content. After content creation, support for an automated conversion into the intermediate xml4tio file format is provided by TIO. Xml4tio is an xml schema definition that is used to store content on a TIO server [25].

Second, formatting instructions stored in xml4tio is reduced to a minimum because small screens can not display sophisticated presentations that were created in Power Point, for example, in a satisfactory manner.

Third, the use of the common storage format has the potential for easier access and for better maintaining a large amount of teaching material, as it is needed for a Bachelor or Master courses, for example.

Fourth, TIO supports multiple end-user devices and operating systems by engaging browsers as the only needed software at the end-user device.

Fifth, TIO supports the learning process not only for full time students, but also for part time customers, together with training on the job for life-long learning of professionals. Therefore, it can be used by universities and enterprises.

Sixth, the teaching material is structured by TIO into ‘learn objects‘ which can be compared to book chapters but may contain audio, video, text, graphics, formulas, visualization of simulation and access to remote laboratories, as well as exercises and multiple choice tests for each topic.

Seventh, TIO can administer a large number of students as customers and a large number of courses as well because it uses ILIAS which has proven to be a reliable system.

Eighth, most of the TIO portals are multilingual, currently in German and English, which gives also a transnational aspect.

4 TIO Set-up

TIO’s set-up is based on three (virtual) servers, one for authors, one for students, and one for backup (**Error! Reference source not found.**).

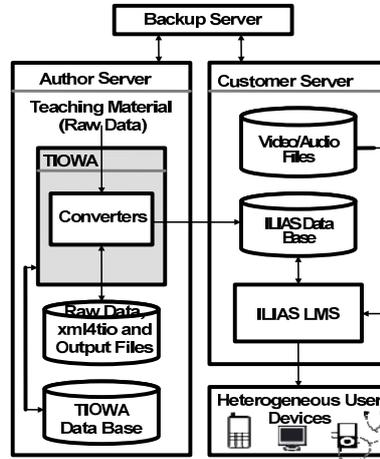


Fig. 1. General hardware setup of TIO

All servers use UBUNTU Linux. On the author’s server, our TIOWA web application is the main software component, together with several converters between storage formats. TIOWA is the frame in which authors create, maintain and convert learn objects. Additionally, a relational data base is provided that cares for authentication and authorization of authors and system administrators, and that maintains different versions of teaching objects. Teaching objects are stored as files in xml4tio. In **Error! Reference source not found.**, the software set-up of the author server is shown. On the customers server, ILIAS is the software which cares for student administration and examination, inter-student communication and content dissemination. In **Error! Reference source not found.**, we see the block diagram of the customer server.

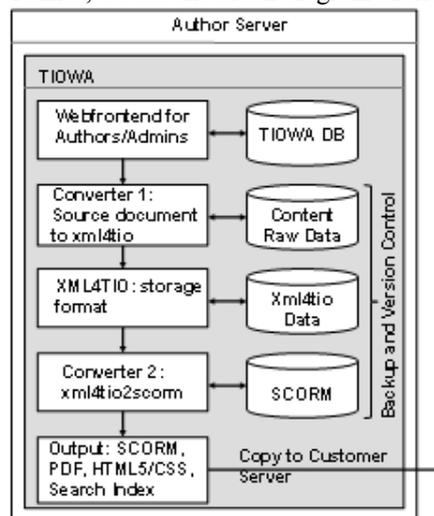


Fig. 2. Set-up of the customer server

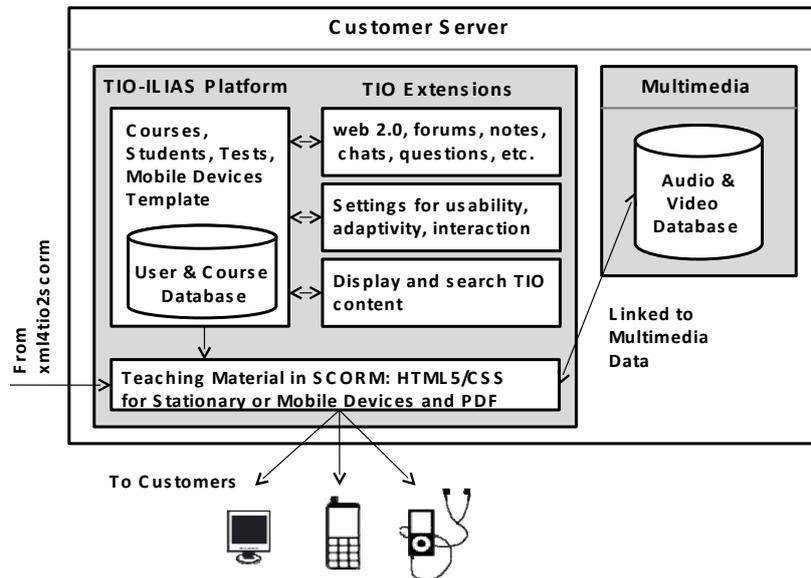


Fig. 3. Set-up of the customer server

The information which student is enrolled in which course is stored in a MySQL data base. Audio and video recordings of teaching objects and whole lectures are stored as standard files with references to xml4tio and SCORM data structures. SCORM and xml4tio are both based on xml. SCORM is a collection of specifications and programming interfaces. It stores user preferences, records learning goals, logs learn progress and describes which resources a learn object has. A resource is a set of texts, pictures, audios, videos and URLs that is organized as a tree. Each tree resembles the chapters and subchapters of a traditional lecture of 45-90 minutes and can be augmented by meta data that describe the learn object. This meta data allows to search for content. Furthermore, possible sequences can be specified in which students can consume learn objects. Finally, the path through a sequence of teaching objects can be declared as a function of the answers the user gives in multiple-choice tests. The backup server provides for a safe operation by copying automatically data from the author and customer servers. In the following, all described components are explained in more detail.

4.1 The TIOWA Web Application

For content creation, management and format conversion, TIOWA was programmed in PHP 5.4 and Javascript as a web application for the authors. It uses MySQL 5.6 [19] as data base for authors and administration, jQuery and Ajax for easier communication with the user via dynamic web pages, and an Apache web server for page generation in HTML5 and CSS. TIOWA allows to upload content from an author's computer to the TIO server, to convert it into xml4tio, to update and backup it, to

move it to the customer server and, in the final step, to reformat the content for display, depending on the respective output device. For this purpose, TIOWA accesses xml4tio data, author/admin data and SCORM data. From the viewpoint of xml4tio, TIOWA is an application for single source publishing.

4.2 XML4TIO for Single Source Publishing

Xml4tio is a xml schema definition [24] and the core of TIO. For the conversion of teaching material into xml4tio, two files and one extra folder must be created: First, a 'container.xml' file is established that stores a description of the teaching material as meta data. In case of B/M modules, the meta data are based on the Bologna module description [20]. Otherwise, the author must provide an arbitrary text as abstract. Additionally, container.xml binds together all teaching objects into lectures and all lectures about the same topic into one Bologna module. Second, a 'content.xml' file must be created that stores the content of the Bologna module.

The first layer of the syntax tree of xml4tio comprises the xml tags module, title, author, section and presentation unit. These tags can subsequently be specialized by additional tags and/or attributes in more layers. Especially remarkable is for example the 'media' tag that can have as attribute 'picture', 'animation', 'applet', 'scene3d', 'sound', 'video' and 'experiment'. This offer shows the capability of TIO in media presentations for MINT subjects.

4.3 Converters for Storing and Disseminating Teaching Material

There are converters from OO Writer, Adobe Frame Maker and LearnDSL into xml4tio. A reverse conversion for a so-called round trip is normally not possible because nearly all character and chapter formatting instructions are deleted during the conversion process. However, if the author limits himself in his text system on the few formatting data that xml4tio has then he can also perform a roundtrip.

Additionally, there exists a converter for transforming xml4tio into SCORM, html5 and pdf. These formats contain less formatting information than the authors' original documents.

The main difficulty the converters are facing with stems from the fact that humans normally do not treat their lecturing materials as a formal text. This means, that beside the used character and chapter formats, several hand-made changes exist in real-world lectures. Such documents can not be converted. To detect these flaws, a xml4tio validator is provided that checks whether the lecture is structured according to the xml4tio schema definition.

Conversion of Teaching Material into XML4TIO. For the LearnDSL converter, own software in Java was written. For the OO Writer, a PHP software was created that transform valid OO texts into xml4tio. Prerequisite for that is that a prescribed LearnDSL and OO character and chapter format catalogue is used without any manual additions or modifications. Furthermore, specific rules have to be obeyed with respect to the structure of the teaching material.

For the Frame Maker (FM) converter, basically the same restrictions hold as for the

Writer converter. However, no software has to be programmed as in the LearnDSL or OO case. FM must only be set from unstructured to structured mode. Then it can directly deliver the desired xml schema. FM uses as input for this schema a so-called Element Definition Document (EDD) file [21]. This means that the EDD let Frame Maker know which Elements are allowed and which composition of elements are legal. The EDD additionally prescribes how to format them elements. By means of a proper EDD, a structured text appears to the user of FM nearly as an unstructured text.

The EDD in turn can be created out of a so-called document type definition (DTD) [22] in a two-step manner: first the DTD file must be manually created on basis of the xml4tio schema definition. Then, the DTD can be imported into FM, and an EDD is created automatically by FM.

However, the resulting EDD file must be post processed manually to define the appearance of chapter and character formats of the elements. Such format definitions are made by templates. The manual post processing of the EDD file is also needed to provide for so-called read/write rules which help to convert FM documents into correct xml4tio by transforming FM elements into TIO tags and attributes.

After that, the EDD file must undergo an automatic transformation by XSLT to create proper URLs, names and paths for pictures and other multimedia content according to the xml4tio schema definition. Therefore, an XSLT style sheet was created, and additional EDD rules were defined that call the style sheet.

Finally, FM bundles the DTD together with the EDD that contains format templates and read/write rules into a so called structured application definition. As soon as such a definition exists, the user can just save any structured FM document into xml4tio by simply clicking the "save as XML" button, and the conversion is done.

Conversion of XML4TIO to SCORM Output. This converter is called 'xml4tio2scorm'. It is responsible for content presentation because the created SCORM file contains html for a browser, together with CSS formats and links to multimedia data such as pictures and graphics. However, the full potential SCORM has is not needed here. Only a TIO-specific subset is used that is subsequently stored in ZIP format in order to save (virtual) disk space. xml4tio2scorm creates a table of content for the selected study course, its html/CSS and pdf representation and a index for searching. This index is copied automatically as a file into the user space of ILIAS so that students can access it.

Depending on the fact whether the target browser is located in a mobile or stationary device and with respect to the device's screen size, either the table of content is presented simultaneously together with a video recording of the lecture and the text of the teaching material. Or in case of small screens, only one of these three streams is displayed, according to the user wishes. Additionally, it is taken into account that mobile devices may have only limited bandwidth for Internet access. Because of that, offline browsing of previously downloaded teaching material is supported, together with pdf viewing.

4.4 Extensions to ILIAS

The following extensions to ILIAS were made: 1.) students can highlight and comment every line of the teaching material. These comments can be attributed to be either private, visible to the public or open for discussion in a special forum. This makes students to prosumers. 2.) students can ask the author of the teaching material via the Internet by placing their question directly into the script at the proper line which eases the communication between student and teacher/author significantly. 3.) Searching in the teaching material is possible by means of an index. 4.) Fonts, font sizes and colors can be configured individually to provide for better reception. 5.) A table of content can be displayed that helps to get a better overview of the learning material 6.) The SCORM output is adapted to mobile devices with individual screen sizes.

The extensions make students into “prosumers” that contribute to their learn success by own comments and hints in the teaching material that are visible to all others. The improved questioning and chat room feature allows a „learning adventure“ and makes TIO into an Internet-based social media.

5 Software Tests and Practical Experiences

For testing the software we received from Clausthal University of Technology three virtual servers which were also administrated by them, including backup and restart in case of system crash. On these servers, three web portals were created: 1.) A portal under <http://webadmin.ti-online.org> for accessing the TIOWA web applications. This portal is for authors and admins. 2.) A portal for general information about the TIO project and for project protocols under <http://ti-online.org>. 3.) A portal for TIO students and other customers to download learning content under <http://ilias.ti-online.org/>. This portal is also for teachers to manage TIO students and courses. It is based on a modified version of ILIAS as described. All portals are maintained by Typo3 [23] as content management system, the last portal is in German and in English language.

5.1 End-User Devices and Configurations

TIO was tested with the following end-user devices and configurations: 1.) desktop PCs and notebooks with MS Windows XP/7/8 as operating systems, and with Firefox (>V2), Internet Explorer (>V6), Opera (>V8), Chrome (>V16) and Safari (>V5) as browsers. 2.) desktop PCs and notebooks with Ubuntu, Debian and Suse Linux together with Opera and Firefox. 3.) Apple computers with Mac OS X (>V10.6), Safari and Firefox. 4.) Apple iPad with iOS (>V5) and diverse Android Tablets (>V4.0). 4.) Apple iPhone (V4) and diverse Android Smartphones (>V2.0), and 5.) Blackberry and Apple iPod. This was considered a comprehensive selection.

5.2 Practical Experiences

In the years 2010-2013, teaching materials were created for TIO in Frame Maker, LearnDSL and Open Office Writer. This happened for the education in Technical Informatics, Computer Organization and Computer Networks at the Universities of Hamburg and Clausthal. Additionally, some video recordings of the lectures, simulations and animations are added to the teaching materials. After every semester, feedback from the students was collected in written form and evaluated. According to that, parts of the software and the didactic presentation of the teaching material were improved.

6 Conclusion and Future Work

A web-based software platform called TIO for mobile learning with emphasis on study courses in mathematics, informatics, natural sciences and technology was created. It was tested for numerous software and hardware configurations users may have and proved to be technically working. TIO consists of a modified version of the open-source E-learning system ILIAS and the TIO tool set. The ILIAS extensions were made to improve its usability for mobile devices and MINT subjects. While ILIAS is for students, TIOWA is for the authors for content creation, uploading and converting. It stores the teaching material in a xml schema definition out of which several output streams can be generated that depend on the end-user device and its screen size (single-source-publishing). Teaching material can contain texts with formulas, audios and videos, animations and visualization of simulation results. Finally, the xml4tio data structure also allows remote access to experimental labs located in a university for practical training. We believe that some of our implemented features will be useful for the generation of 'digital natives' that must prosecute life-long learning.

The general experience is that mobile learning is useful for MINT subjects provided that diverse end user devices are supported because they differ a lot in their capabilities. Furthermore, it proved to be important to support various text systems such as Frame Maker, Open Office Writer and a variant of LaTeX because otherwise not enough authors can be found to develop teaching material. Furthermore, we found that mobile learning is especially useful for professionals because they can learn now in a flexible way in empty times slots such on train trips between work and home. Finally, an emotional component should be existent in mobile learning that makes it a deeper and more lasting. Thus chat rooms and other features with which clients can communicate with each other in the style of a social media were integrated. As a recommendation, we suggest to combine mobile learning with classroom presence of approx. 10% of the study time to make the emotional component optimal.

In the future, TIO must be approbated, tested and checked in practice over a longer period of time. Therefore, it will be used for online teaching a Master course in Technical Informatics and for training and certifying of professionals for the purpose of life-long learning. These real-world applications will allow to conduct a pedagogical evaluation of the benefits and disadvantages of M-learning for MINT subjects. From

that experience, recommendations can be given to teachers and organizations about M-learning in general. The TIO tool-set is the technical prerequisite for that.

References

1. Adobe Framemaker, <http://www.adobe.com/products/framemaker.html>
2. Open Office Writer, <http://www.openoffice.org/>
3. ILIAS Learning Management System, <http://www.ilias.de/>
4. Holzinger, A., Nischelwitzer, A., Meisenberger, M.: Lifelong-Learning Support by M-Learning: Example Scenarios, eLearning Magazine, 11, ACM, New York (2005)
5. Guy, R.: Mobile Learning: Pilot Projects and Initiatives, Informing Science Press (2010)
6. Kitchenham, A.: Models for Interdisciplinary Mobile Learning, Delivering Information to Students, IGI Global (2011)
7. Sampson, D. G., Isaias, P., Ifenthaler, D., Spector, J. M.: Ubiquitous and Mobile Learning in the Digital Age, Springer New York, Heidelberg, Dordrecht, London. (2013)
8. http://en.wikipedia.org/wiki/Digital_native
9. http://www.techworld.com.au/article/223565/10_open_source_elearning_projects_watch/
10. PHP - Hypertext Preprocessor, Web Scripting Language, <http://php.net/>
11. Sharable Content Object Reference Model (SCORM), <http://scorm.com/scorm-explained/>
12. Ajax (Asynchronous JavaScript and XML),
13. Javascript, <http://www.w3.org/standards/webdesign/script.html>
14. Introduction in Extensible Markup Language (XML), <http://www.w3.org/XML/>
15. World Wide Web Consortium. Standards, <http://www.w3.org/standards>
16. World Wide Web Consortium. Cascading Style Sheets, <http://www.w3.org/Style/CSS/>
17. World Wide Web Consortium. XSL Transformations, <http://www.w3.org/TR/xslt>
18. World Wide Web Consortium. XML Schema, <http://www.w3.org/XML/Schema.html>
19. MySQL 5.6 Reference Manual, <http://dev.mysql.com/doc/refman/5.6/en/index.html>
20. Bologna process, <http://ec.europa.eu/education/policies/educ/bologna/bologna.pdf> and <http://www.ehea.info/>
21. Element Definition Document, http://help.adobe.com/en_US/FrameMaker/8.0/help.html?content=Chap2-FrameMaker-Basics_098.html
22. Document Type Definition, http://help.adobe.com/en_US/FrameMaker/8.0/help.html?content=Chap2-FrameMaker-Basics_098.html
23. Typo3 - The Enterprise Open Source CMS, <http://typo3.org/>
24. Moodle - <https://moodle.org/>
25. XML4TIO, <http://www.ti-online.org/XML4TIO>

Holistic Approach to Training of ICT Skilled Educational Personnel

Mariya Shyshkina

Institute of Information Technologies and Learning Tools
of the National Academy of Pedagogical Sciences of Ukraine

marple@ukr.net

Abstract. The article intends to explore and estimate the possible pedagogical advantages and potential of cloud computing technology with aim to increase organizational level, availability and quality of ICT-based learning tools and resources. Holistic model of a specialist is proposed and the problems of development of a system of methodological and technological support for elaboration of cloud-based learning environment of educational institution are considered.

Keywords. Learning environment, personnel training, cloud computing, holistic approach

Key terms. KnowledgeEvolution, KnowledgeManagementMethodology, Didactics, KnowledgeManagementProcess, ICTInfrastructure

1 Introduction

As it is now impossible to introduce advanced ICT while managing this process without mastering the ICT and other related pedagogical technologies, the main aim is to train ICT-skilled educational personnel. Cloud computing technology (CC) is to create a high-tech learning environment of educational institution, enhancing multiple access and joint use of educational resources at different levels and domains. On this basis it is possible to combine corporate resources of the university and other on-line resources, adapted to learning needs, within a unite framework.

Cloud computing is used for resources supply and to support collaboration in the learning process in particular by means of mobile services. It requires the development of new approaches and models for designing of a learning environment. Among them there are those based on a holistic approach to learning [1], [5], [7], [9].

For this aim a set of instrumentation tools for cloud-based learning resources collection, elaboration and design, holistic models of learning environment and specialist models, and a system of methodological and technological support for the development of cloud-based learning environment of educational institution should be created.

The *purpose of the article* is to identify trends and conceptual models of educational personnel training within the cloud based learning environment.

2 Problem Statement

The problem of training of qualified educational management personnel as well as teachers oriented on ICT based learning can nowadays hardly be taken independently from the processes of the innovative development of educational space formed within the school, region and educational system of a country or globally [1]. In this regard, there is a need for fundamental research focusing on the possible ways of developing an educational environment of educational institutions. It should take into account, the trends of improving ICT facilities while searching for new engineering technological decisions and new pedagogical and organizational models [1], [2]. The main focus is on shifting from mass introduction of separate software products, to an integrated and combined environment which supports distributed network services and cross-platform solutions.

Emerging technologies of information and communication networks give a way for implementing a holistic approach to education and training of personnel. A holistic approach focuses on combining science and practice, training and production, fundamental and applied knowledge and technological competencies with social and humanitarian [5], [9]. Above all it aims the development of public administration's management skills in the educational field basing on a unite approach to learning design and management. This is a promising direction for the development of a field's human potential. The innovative processes therefore, of the organization and development of learning environment, search for new approaches and models for specialist education and training becomes a matter of interest [11].

There is **a problem** of availability and valuable ways of learning resources delivery, to achieve with their use the best pedagogical effect and to gain maximum learning potential of ICT. This issue, may be hence partially solved if delivered by means of cloud computing technology [2], [8], [12]. The main advantage of this technology is the improved access to qualitative resources (and sometimes the only possible access to necessary recourses at all). **The idea** is simply to explore approaches for the modeling and estimation of CC-based learning process settings and valuable tools for its organization.

3 Education and Training of ICT-Skilled Management and Public Administration Personnel

Public administrators are public servants working in public institutions, departments and agencies [10]. Specifically, they are concerned with "planning, organizing, directing, coordinating, and controlling government operations" [6]. Specific sphere are public servants for education management. For such personnel to be efficiently trained, the need to develop novel approaches arises, as this sphere is mostly concerned with multi-disciplinary knowledge and requires skills on the merge of training,

learning and management. Due to the fact that most pedagogical innovations are also based on ICT the need in the sphere of education management also arises. There is a branch of pedagogical sciences dealing with theoretical and methodological problems of ICT in education use, psychological and pedagogical substantiation of these processes, elaboration of ICT tools and resources for providing functioning and development of educational systems. So there should be specialized personnel to insure the processes of implementation, introduction and development of ICT-based learning technologies within the sector of public administration.

There are significant needs in IT competent specialists in the sphere of public administration. Without ICT competence or competence in ICT for learning, problems with their adaptation at the workplace arise, as do problems with the necessity of additional and often profound training almost immediately after hiring. In some cases, a vague idea of future graduates about the real problems and conditions of work with innovative ICT infrastructures and ICT-based tools leads to lack of commitment to practical solutions of work situations thus to a low level of innovative inclusion.

Formation of the innovative institution's ICT infrastructure could solve some of the aforementioned problems [11]. Namely, it would bridge the gap between the process of training and the level of demand for their product. An environment that would bring together the learning resources of educational and industrial projects would be created, and would cover different levels of training; including the training of both students and pedagogical management personnel.

According therefore to the high rates of development of both the global ICT market for the education sector, and the IT market of learning tools, the problem of training professional staff for the domestic public administration and the IT-oriented sector of education management; personnel which are primarily prepared within higher and post graduate schools (e.g. universities and advanced training schools) being continuous, we conclude that modern approaches to the design of educational systems are a key point.

It is unlikely that the current state of skilled personnel of management and public administration of education could be regarded as fully satisfactory for the needs of innovative development of ICT-based learning, for the required number of qualified professionals with appropriate structure and quality of training. The system of training and retraining of employees for public administration has not been properly formed.

These problems should be considered within the context of development of an institution's and a region's innovative environment as well as on national and international level [3], [11]. These processes have to do with the modernization of a learning environment in perspective of emerging ICT. Thus the developmental need of new models and approaches to personnel training arises, which will account for the modernization of ICT infrastructure and integrate resources of different levels and use.

Introduction of innovations into the educational environment of a state or a region, is highly concerned with the development of human resources of informatization on education [1]. It requires new types of skills and competencies which graduates often lack of. These skills include leadership, ability to approach a problem holistically, and the ability to critically evaluate achievement and self-assessment [9], [11]. It is the lack of qualified personnel and the absence of a strategic approach to ICT infrastruc-

ture design that are among the reasons for an institution's of professional education deficiency of a unite high-tech desisions.

Nowadays content-technological process regarding the creation and use of ICT products, and in particular the electronic learning resources, requires fundamental background knowledge in both ICT and pedagogy. The approaches however for training personnel today, do not sufficiently take into account the recent years' innovative changes in the ICT industry, nor the real needs regarding the extent of such training.

A mean for provision of users with relevant services of cloud computing technology is considered to be outsourcing; i.e. a service in a specific system to implement its core functions is required, offered and sold by another system external to this [3]. ICT outsourcing plays an important role in enhancing the scientific and technical level of ICT-systems of an educational institution as well as the efficiency of their operation and their development. It is a market mechanism incorporating the latest advances in the ICT sector and to satisfying user demand [2], [3].

The main problem in educational practice is the contradiction between on one hand the objective need for a continuous improvement of the software and the hardware power of training computer complexes, and on the other, the lack of personnel's ability (in both qualitative and quantitative manners) to maintain, manage and develop their ICT systems appropriately. The informatization hence of an educational institution in terms of cloud computing and ICT outsourcing, will offer realistic solutions for both the deepening of informatization and improvement of ICT's educational performance and use of information resources [2], [3].

The basic principles of such introduction should be: a tight relationship of learning with training and methodological support for tutors, focus on a specific educational task; modularity of learning; continuity of learning, sharing experience and formation and participation in professional association activities (including electronic) [2], [3]. In this process electronic distance learning systems should be actively used, based on the principles of open education, with the maximum possible use being of CC technology and outsourcing.

4 What are Advantages of Cloud Computing Decision?

4.1 It is a Cost Effective Solution

Being cost-effective the user can get (buy) products and services proposed by the virtual supermarket of ICT according to their needs (individual or group, collective, corporate), they may pay only for what has been bought (e -transport, e-content, e-services, virtual e-tools, a generic and subject software applications, network platforms - full range of cloud services along with services for the design and implementation of ICT systems and their fragments ordered by the users, their warranty and post warranty service, maintain, upgrade and improvement, etc.) and only for the actual time of use of the purchased product [3]. This will allow users to avoid regular updating and upgrading of powerful general system software and hardware tools of their own ICT systems, avoiding a potential surplus of ICT products used from time

to time; fragmentary, not fully, as well as spare parts, reduction of requirements for information security of their own ICT systems, reduction of the number of their ICT services and requirements for professional competence of their employees and as a result, significantly reduce overall costs to support the operation and develop their ICT systems, to increase their social and economic return, their efficiency [2], [3].

4.2 This is a Flexible Solution of ICT Infrastructure

It is designed for increased flexibility and effective access to learning resources so as to build a unified and mobile infrastructure.

On the basis of CC infrastructure all main aspects of interaction of a learner may be comprehended on the unite basis. Along to approach introduced in [1], among them there will be interactions between a learner and other learners; a learner and a teacher; a learner and a learning tool; a learner and educational institution; a learner and the society. This will lead to an environment of learning organization on the unite base, where collaboration between learners and a tutor, free and flexible resource access, learning activity within social inclusion into the environment of an educational institution and the society will be enabled. The ICT support of learning is realized by means of cloud services. It is designed for adaptation to the rapidly changing external/internal environment, changing of task/competence requirements and development of modern pedagogical approaches.

Due to the principles of open education [1], there is a need to create an innovative learning environment that will form and develop necessary professional skills. Among them are leader skills, collaborative skills, critical thinking, and the ability to view a problem in a holistic manner. These skills refer mostly to the demand of the sphere of public administration of education, as in alliance with them; a process of innovative development may be involved. This may be achieved on the basis of a holistic approach to specialist training when the planning, design and resource management and learning activity of an organization and its monitoring, may be represented on a unite basis. It will be achieved through the unite development of different competencies: professional, fundamental, personal and technological.

5 A Holistic Model of a Specialist

Holistic approach to education deals with the learning processes to be taken as unity of all main aspects of a personality development, for example such as mental, emotional and volitional. This is in tune with a meaning of the term “holistic” as completeness, being impossible with disregard of some of its components.

There are innumerable investigations devoted to the problems of holistic learning development in different aspects such as learning and teaching interaction, collaboration processes, engagement of both aspects of theory and practice to gain comprehensive view of a subject [7], [9]. Now there are important trends of research development in concern to modern ICT. For example, holistic view is to approach learning environment structure. Thus, the model of a learning environment, developed in [1] is

to reveal main components and types of interactions within the different learning process settings.

The notion of holistic learning occurs in relation to personnel training, concerning to different components and interactions within educational organization. It may touch upon certain types of activity, collaboration and resource management processes, engaging thus the entire organization at all levels and developing a performance culture of personnel. There are different ways to approach peculiarities of specialist formation, namely in the aspect of personal or professional features. That concerning to modeling of professional competencies [5], especially in the sphere of educational management. Another aspect is about holistic models to develop leader skills [4], [9], which are more to traits of a personality.

The proposed approach is based on holistic model of a specialist in the sphere of informatization of education presented in Fig.1. It concerns to Domain Competencies which would occupy fundamental knowledge of educational management and modern learning technologies and also ICT skills and ability to use e-learning tools. There are also Personal Competencies, such as leader skills, critical thinking, and capability to holistic view of a problem, responsibility and activity of an individual. As for professional skills there are planning, design, resources management, cooperation and collaboration skills, performance skills and ability for monitoring and self evaluation.

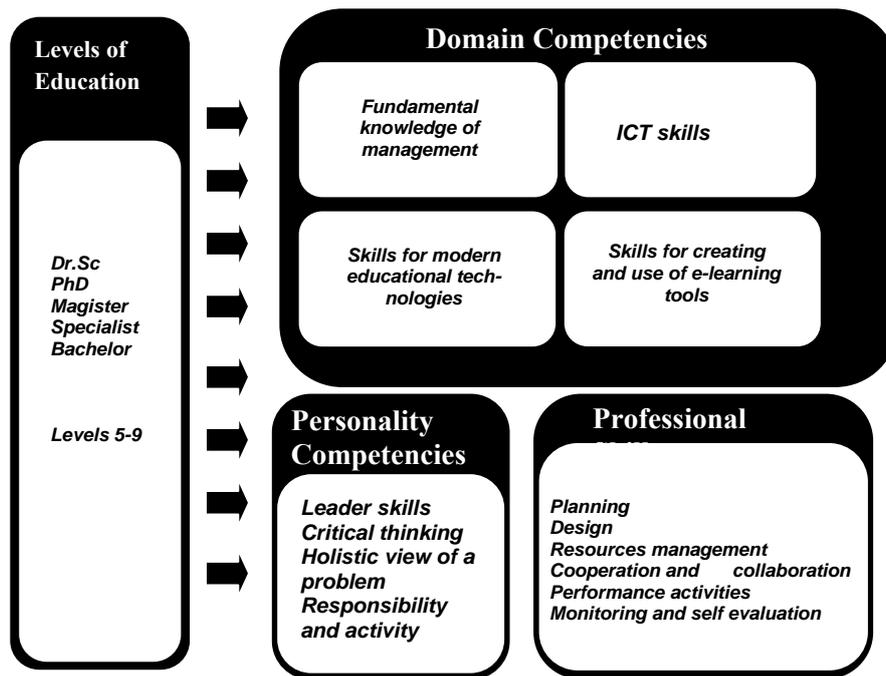


Fig. 1. A holistic model of a specialist

All the components of a specialist's competencies, skills and knowledge are consistently formed within the main level of education which corresponds to National qualification framework (levels 5-9).

Cloud computing decision is a reasonable way to support holistic learning settings giving a platform for unite representation and access to learning tools for different levels and domain of education as also for different individuals and groups of users.

Promising ways of assessing resources quality, while building a holistic learning environment are:

- A. Analysis of the most appropriate ways to use cloud computing technology to supplementing and structuring collection of educational learning resources, filling it with the resources on this basis and organizing multiple access to their use
- B. Use of a certain set of educational resources for testing methods to evaluate the quality of their use within the cloud-based infrastructure of organization
- C. Recommendations on methods to replenish the collection, its prototyping and ways of structuring resources
- D. Elaboration of requirements to provide electronic resources, for collection replenishment
- E. Analysis of cloud computing technology outsourcing for optimal selection and use of resources' collections
- F. Creation of recommendations to developers and material for replenishment and application of existing electronic learning resources
- G. Development of recommendations for dissemination and use of collections of electronic resources

6 An Expected Impact and Social Results of the Project

The important step to wider application and introduction of new learning approaches and to gain most possible benefit from emerging technologies and ICT tools should be achieved through modernization and upgrading of ICT learning environment of educational institutions, increasing of overall level of e-learning.

To achieve these goals the main problem is to rise ICT and professional level of competencies of subjects of the learning process – managers, pedagogical personnel and staff and also personal of ICT departments. Just the people are the most valuable factor of empowerment of development and formation of social and economical systems and educational systems in particular. Just the people are the most important resource which should be involved so as to improve the quality of these social systems and to manage their purposeful and productive growth. By this reason development of tools and resources to train teachers and staff is critical point because it really concern to all levels of educational systems functioning.

The whole impact of implementation of learning tools and techniques based on cloud computing is aimed at:

- Broaden use of ICT in education aiming at wider take up by learners and teachers
- Effective public-private partnerships for introduction and managements of learning environment solutions

- More efficient introduction of ICT into the learning process through the exploitation of monitoring and assessment tools
- More timely and purposeful acquisition of skills and competences through ICT-based learning technologies, in educational establishments and public administrations
- Increased involvement with the adoption of learning digital technologies

The important step to wider application and introduction of new learning approaches and to gain most possible benefit from emerging technologies and ICT tools should be achieved through modernization and upgrading of ICT learning environment of educational institutions, development of new learning approaches, creating more advanced learning technologies [1], [2].

Formation of innovative ICT infrastructure of the institution could solve some of the problems of development highly skilled educational and management personnel, bridging the gap between the process of training and the level of demand for their product.

Due to development of cloud computing technologies opportunities, functionality and access to collections of electronic learning resources has significantly increased. In this regard, cloud computing is a promising direction of development of electronic resources' collections (may be relevant for development of collections), as it allows the creation of a unified methodology for a single platform, a framework for development and testing, and for improvement and elaboration of integrated assessment methods' quality. This gives an added value to available recourses [2], [11].

The **social results** will help to modernize the learning environment of educational institutions and organizations, to increase educational potential of ICT and add value to the best examples of available learning resources due to their flexible and learner-adaptive access.

At the same time there are several aspects of the cloud-based learning architecture to be a subjected to further research. There are problems of pedagogical and psychological support in regard to the processes of the design and organization of an educational institution's cloud infrastructure, prospecting possible organizational structures to provide learning environment functioning and to teach educational managers and organizers, pedagogical and technical staff how to use new methods and approaches to learning, based on cloud computing. There is a necessity therefore, to create an educational and training system of support used by management personnel, teachers and learners.

The result of instrumentation for cloud-based learning resources collection elaboration, and development of cloud-based learning environment of educational institution might be used within different learning and organizational educational structures.

7 Analysis and Estimation of Perspective Ways of Development

The cloud based learning infrastructure is to give the opportunities:

- To combine the processes of development and use of electronic resources to support learner competencies

- To insure holistic approach to specialist education and training, combining both technological and social competences, development of critical skills of a learner
- To integrate the processes of training, retraining and advanced training, at different levels of education by providing access to electronic resources of a unite learning environment
- To solve or significantly mitigate the problems of association of electronic resources of the institution into unite framework
- To access to the best examples of electronic resources and services to those units or institutions, where there is no strong ICT support services for e-learning
- To provide of invariant access to learning resources within the unified educational environment, depending on the purpose of study or educational level of the student, enabling person-oriented approach to learning
- To make conditions for a higher level of harmonization, standardization and quality of electronic resources, which may lead to emergence of the better examples of learning resources and to more massive use them

8 Conclusion

There are real advantages of CC technologies to assure more flexible, scalable and cost-effective decisions of access to learning resources as within the learning environment of the university and also in learning environment of the whole region, national and international scale. This is an advantage so as to ensure joint use and widening participation in the learning courses of learners from different institution were necessary services are substantiated and supported. As if holistic approaches to cloud services development are already used in education so the challenge is to transfer this experience into wider context.

The project is implemented within the framework of the joint research laboratory of Cloud computing in education of the Institute of Information Technologies and Learning Tools of NAPS of Ukraine (Kiev) and the Krivoy Rog State University (Krivoy Rog), www.ccelab.ho.ua.

References

1. Bykov, V.: Models of Organizational Systems of Open Education. Atika, Kyiv (2009) (in Ukrainian)
2. Bykov V.: Cloud Computing Technologies, ICT Outsourcing, and New Functions of ICT Departments of Educational and Research Institutions. *Information Technologies in Education*, 10, 8–23 (2011) (in Ukrainian)
3. Bykov V., Shyshkina M.: Innovative Models of Education and Training of Skilled Personnel for High Tech Industries in Ukraine. *Information Technologies in Education*, 15, 19–29 (2013)
4. Candis Best, K.: Holistic Leadership: a Model for Leader-Member Engagement and Development. *The Journal of Values Based Leadership*, 4(1) (2011)
5. Cheetham, G., Chivers, G.: Towards a Holistic Model of Professional Competence. *Journal of European Industrial Training*, 20(5), 20–30 (1996)

6. Chapman B., Mosher F. C., Page E. C.: Public Administration. Encyclopedia Britannica, <http://www.britannica.com/EBchecked/topic/482290/public-administration>
7. Forbes, S. H., Martin, R. A.: What Holistic Education Claims About Itself: an Analysis of Holistic Schools' Literature. In: Proc. Annual Conf. American Education Research Association, San Diego, California (2004)
8. Zhang, Qi, Cheng, Lu, Boutaba, R.: Cloud Computing: State-of-the-Art and Research Challenges. *J. Internet Serv. Appl.*, 1, 7–18 (2010)
9. Quatro, S. A., Waldman, D. A. Galvin, B.M.: Developing Holistic Leaders: Four Domains for Leadership Development and Practice. *Human Resource Management Review*, 17, 427–441 (2007)
10. Kettl, D.; Fessler J.: *The Politics of the Administrative Process*. CQ Press, Washington D.C. (2009)
11. Shyshkina, M.: Innovative Technologies for Development of Learning Research Space of Educational Institution. *Information Technologies and Society*, 1, http://ifets.ieee.org/russian/depository/v16_i1/pdf/15.pdf (2013) (In Russian)
12. Sultan, N.: Cloud Computing for Education: A New Dawn? *Int. J. of Information Management*, 30, 109–116 (2010)

**2.3 2nd International Workshop
on Algebraic, Logical, and Algorithmic
Methods of System Modeling,
Specification and Verification
(SMSV 2013)**

Foreword

It is our pleasure to offer you the selection of papers for the 2nd International Workshop on Algebraic, Logical, and Algorithmic Methods of System Modeling, Specification and Verification (SMSV 2013) which has been co-located with the 9-th International Conference on ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer (ICTERI 2013) held at Kherson, Ukraine on June 19-22, 2013.

Workshop SMSV 2013 is a successor of the International Workshop on Algebraic, Logical, and Algorithmic Methods of System Modeling, Specification and Verification (SMSV 2012) which was held in Kherson (Ukraine) on June, 6-12, 2012. The SMSV 2013 was organized by Kherson State University, Taras Shevchenko National University of Kyiv, and Paul Sabatier University of Toulouse within the framework of the cooperation agreement between the universities. The workshop attracted scientists from Austria, France, Algeria, Russia, and Ukraine.

Presented papers demonstrated the interest in the topics on different formal methods of system development, and we plan to organize such workshops on a regular basis. We hope that presentations and discussions will help to identify topics of mutual interest that can be considered as a base of project proposals to be submitted to international scientific programs.

June, 2013

Vladimir Peschanenko
Mykola Nikitchenko
Martin Strecker

An Abstract Block Formalism for Engineering Systems^{*}

Ievgen Ivanov^{1,2}

¹Taras Shevchenko National University of Kyiv, Ukraine

²Paul Sabatier University, Toulouse, France

`ivanov.eugen@gmail.com`

Abstract. We propose an abstract block diagram formalism based on the notions of a signal as a time-varying quantity, a block as a signal transformer, a connection between blocks as a signal equality constraint, and a block diagram as a collection of interconnected blocks. It does not enforce implementation details (like internal state-space) or particular kinds of dynamic behavior (like alternation of discrete steps and continuous evolutions) on blocks and can be considered as an abstraction of block diagram languages used by engineering system designers.

We study its properties and give general conditions for well-definedness of the operation of a system specified by a block diagram for each admissible input signal(s).

Keywords. block diagram, signal transformer, semantics, engineering system

Key Terms. Mathematical Model, Specification Process, Verification Process

1 Introduction

Many software tools for developing control, signal processing, and communication systems are based on block diagram notations familiar to control engineers. Examples include system design software Simulink [1], Scicos [2], Dymola [3], SCADE [4], declarative synchronous languages [5], some embedded programming languages [6, 7].

In such notations, a diagram consists of blocks (components) connected by links. Typically, blocks have input and output ports, and (directed) links connect output ports of one block with input ports of the same or another block. A block is interpreted as an operation which transforms input signals (i.e. time-varying quantities) flowing through its input ports into output signals.

^{*} Part of this research has been supported by the project *Verisync* (ANR-10-BLAN-0310), France.

Wide applicability of block diagram notations makes them an interesting object of study from a theoretical perspective. Classical control theory and signal processing already provide some degree of formal treatment of block diagrams [8, 9], but this is normally not sufficient to handle such aspects of modern system design languages as mixing of analog and discrete-time blocks, partially defined block operations, non-numeric data processing, etc.

To take these issues into account, researchers developed formal semantics for various block diagram languages [10–13]. Some effort has been made to unify approaches taken by different engineering system modeling and analysis tools and make them interoperable by the use of exchange languages with well-defined semantics [14, 15] such as Hybrid System Interchange Format (HSIF) which gives semantics of hybrids systems in terms of dynamic networks of hybrid automata [16, 17].

Although hybrid automata-based approaches like HSIF can be used to give semantics to block diagram languages [18] and have many advantages (e.g. availability of verification theory for hybrid automata), we consider them not entirely satisfactory from a theoretical standpoint for the following reasons:

- Semantics of a system component (block) is based on the notion of a dynamical system with an internal state, and so the components with the same externally observable behavior can be semantically distinguishable. In our opinion, this is not needed for a high-level semantics which does not intend to describe details of the component's physical/logical implementation.
- Semantics of a system has a computational nature: it describes a sequence of discrete steps, where a step may involve function computations, solving initial-value problems for differential equations (continuous evolution), etc. It may be adequate for certain classes of discrete-continuous systems, but it does not always capture the behavior of a physical realization of a system (and thus may conflict with the view of a system designer).

For example, a Zeno execution [17] of a hybrid automaton can be described as an infinite sequence of discrete steps which takes a bounded total time (but each step takes a non-zero time). This normally does not correspond to the behavior of a physical system described by the automaton. In many cases this is caused by system modeling simplifications. The conflict is usually resolved by applying a certain method of continuation of an execution beyond Zeno time (regularization, Filippov solution, etc.) [19]. But an extended execution is not, in fact, a sequence of discrete steps, as it resumes after the accumulation point.

In our opinion, in the general case, the dynamic behavior of a system should not be restricted to a particular scheme like a sequence of discrete steps and continuous evolutions.

The goal of this paper is to introduce abstract formal models for blocks and block diagrams which overcomes limitations of the classical control/signal-theoretic approach to them and does not enforce implementation details (like internal state-space) or particular kinds of dynamic behavior (like alternation of discrete steps and continuous evolutions) on blocks.

These models can be used to identify the most general properties of block diagram languages which are valid regardless of implementation details. In particular, in the paper we will give a general formulation and conditions for well-definedness of the operation of a system specified by a block diagram for each admissible input signal(s).

To achieve our goal, we will use a *composition-nominative approach* [20]. The main idea of this approach is that semantics of a system is constructed from semantics of components using special operations called compositions, and the syntactic representation of a system reflects this construction.

The paper is organized in the following way:

- In Section 2 we give definitions of the auxiliary notions which are used in the rest of the paper.
- In Section 3 we introduce abstract notions of a block, a connection, a block diagram, and a block composition. We show how they fit into a system design process and give conditions of well-definedness of the operation of a system specified by a block diagram for each admissible input signal(s).

2 Preliminaries

2.1 Notation

We will use the following notation: $\mathbb{N} = \{1, 2, 3, \dots\}$, $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$, \mathbb{R}_+ is the set of nonnegative real numbers, $f : A \rightarrow B$ is a total function from A to B , $f : A \dashrightarrow B$ is a partial function from A to B , 2^A is the power set of a set A , $f|_X$ is the restriction of a function f to a set X . If A, B are sets, then B^A denotes the set of all total functions from A to B . For a function $f : A \dashrightarrow B$ the symbol $f(x) \downarrow$ ($f(x) \uparrow$) means that $f(x)$ is defined (respectively undefined) on the argument x .

We denote the domain and range of a function as $dom(f) = \{x \mid f(x) \downarrow\}$ and $range(f) = \{y \mid \exists x f(x) \downarrow \wedge y = f(x)\}$ respectively. We will use the the same notation for the domain and range of a binary relation: if $R \subseteq A \times B$, then $dom(R) = \{x \mid \exists y (x, y) \in R\}$ and $range(R) = \{y \mid \exists x (x, y) \in R\}$.

We will use the notation $f(x) \cong g(x)$ for the strong equality (where f and g are partial functions): $f(x) \downarrow$ iff $g(x) \downarrow$ and $f(x) \downarrow$ implies $f(x) = g(x)$.

The symbol \circ denotes a functional composition: $(f \circ g)(x) \cong g(f(x))$.

By T we denote the (positive real) time scale $[0, +\infty)$. We assume that T is equipped with a topology induced by the standard topology on \mathbb{R} .

Additionally, we define the following class of sets:

$$\mathcal{T}_0 = \{\emptyset, T\} \cup \{[0, x] \mid x \in T \setminus \{0\}\} \cup \{[0, x] \mid x \in T\}$$

i.e. the set of (possibly empty, bounded or unbounded) intervals with left end 0.

2.2 Multi-valued functions

A multi-valued function [20] assigns one or more resulting values to each argument value. An application of a multi-valued function to an argument is interpreted as a nondeterministic choice of a result.

Definition 1 ([20]). A (total) multi-valued function from a set A to a set B (denoted as $f : A \xrightarrow{tm} B$) is a function $f : A \rightarrow 2^B \setminus \{\emptyset\}$.

Thus the inclusion $y \in f(x)$ means that y is a possible value of f on x .

2.3 Named sets

We will use a simple notion of a named set to formalize an assignment of values to variable names in program and system semantics.

Definition 2. ([20]) A named set is a partial function $f : V \rightrightarrows W$ from a non-empty set of names V to a set of values W .

In this definition both names and values are unstructured. A named set can be considered as a partial ("flat") case of a more general notion of nominative data [20] which reflects hierarchical data organizations and naming schemes.

We will use a special notation for the set of named sets: ${}^V W$ denotes the set of all named sets $f : V \rightrightarrows W$ (this notation just emphasises that V is interpreted as a set of names). We consider named sets equal, if their graphs are equal.

An expression of the form $[n_1 \mapsto a_1, n_2 \mapsto a_2, \dots]$ (where n_1, n_2, \dots are distinct names) denotes a named set d such that the graph of d is $\{(n_1, a_1), (n_2, a_2), \dots\}$. A nowhere-defined named set is called an *empty named set* and is denoted as \square .

For any named sets d_1, d_2 we write $d_1 \subseteq d_2$ (*named set inclusion*), if the graph of a function d_1 is a subset of the graph of d_2 . We extend set-theoretical operations of union \cup , intersection \cap and difference \setminus to the partial operations on named sets in the following way: the result of a union (intersection, difference) of named sets (operation's arguments) is a named set d such that the graph of d is the union (intersection, difference) of graphs of the arguments (if such d exists).

3 An Abstract Block Formalism

3.1 Abstract block

Let us introduce abstract notions of a signal as a time-varying quantity and a block as a signal transformer. We will use a real time scale for signals, but we will not require them to be continuous or real-valued. So the signals can be piecewise-constant as well and can be used to represent discrete evolutions.

Informally, a block (see Fig. 1) is a device which receives input signals and produces output signals. We call a collection of input (output) signals an input (resp. output) signal bunch. At each time moment (the value of) a given signal may be present or absent. In the general case, the presence of an input signal at a given time does not imply the presence of an output signal at the same or any other time moment.

A block can operate nondeterministically, i.e. for one input signal bunch it may choose an output signal bunch from a set of possible variants. However, for any input signal bunch there exists at least one corresponding output signal

bunch (although the values of all signals in it may be absent at all times, which means that the block does not produce any output values).

Normally, a block processes the whole input signal bunch, and does or does not produce output values. However, in certain cases a block may not process the whole input signal bunch and may terminate at some time moment before its end. This situation is interpreted as an abnormal termination of a block (e.g. caused by an invalid input).

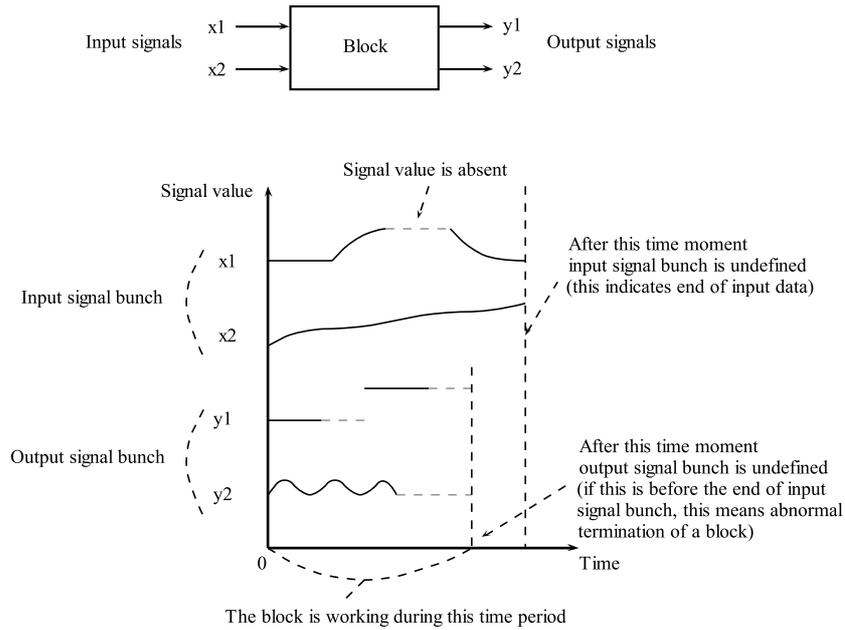


Fig. 1. An illustration of a block with input signals x_1 , x_2 and output signals y_1 , y_2 . The plot displays example evolutions of input and output signals. The input and output signals are lumped into an input and output signal bunch respectively. Solid curves represent (present) signal values. Dashed horizontal segments indicate absence of a signal value. Dashed vertical lines indicate the right boundaries of the domains of signal bunches.

Let us give formal definitions. Let W be a (fixed) non-empty set of values.

- Definition 3.** (1) A signal is a partial function from T to W ($f : T \dashrightarrow W$).
- (2) A V -signal bunch (where V is a set of names) is a function $s : T \dashrightarrow^V W$ such that $\text{dom}(s) \in \mathcal{T}_0$. The set of all V -signal bunches is denoted as $Sb(V, W)$.
- (3) A signal bunch is a V -signal bunch for some V .
- (4) A signal bunch s is trivial, if $\text{dom}(s) = \emptyset$ and is total, if $\text{dom}(s) = T$. A trivial signal bunch is denoted as \perp .

- (5) For a given signal bunch s , a signal corresponding to a name x is a partial function $t \mapsto s(t)(x)$. This signal is denoted as $s[x]$.
- (6) A signal bunch s_1 is a prefix of a signal bunch s_2 (denoted as $s_1 \preceq s_2$), if $s_1 = s_2|_A$ for some $A \in \mathcal{T}_0$.

Note that \preceq on V -signal bunches is a partial order (for an arbitrary V). Later we will need generalized versions of the prefix relation for pairs and indexed families of pairs of signal bunches.

For any signal bunches s_1, s_2, s'_1, s'_2 let us denote $(s_1, s_2) \preceq^2 (s'_1, s'_2)$ iff there exists $A \in \mathcal{T}_0$ such that $s_1 = s'_1|_A$ and $s_2 = s'_2|_A$.

For any indexed families of pairs of signal bunches $(s_j, s'_j)_{j \in J}$ and $(s''_j, s'''_j)_{j \in J}$ of signal bunches let us denote $(s_j, s'_j)_{j \in J} \preceq^{J,2} (s''_j, s'''_j)_{j \in J}$ iff there exists $A \in \mathcal{T}_0$ such that $s_j = s''_j|_A$ and $s'_j = s'''_j|_A$ for all $j \in J$.

It is easy to check that \preceq^2 is a partial order on pairs of signal bunches and $\preceq^{J,2}$ is a partial order on J -indexed families of pairs of signal bunches.

A block has a syntactic aspect (e.g. a description in a specification language) and a semantic aspect – a partial multi-valued function on signal bunches.

- Definition 4.** (1) A block is an object B (syntactic aspect) together with an associated set of input names $In(B)$, a set of output names $Out(B)$, and a total multi-valued function $Op(B) : Sb(In(B), W) \xrightarrow{tm} Sb(Out(B), W)$ (operation, semantic aspect) such that $o \in Op(B)(i)$ implies $dom(o) \subseteq dom(i)$.
- (2) Two blocks B_1, B_2 are semantically identical, if $In(B_1) = In(B_2)$, $Out(B_1) = Out(B_2)$, and $Op(B_1) = Op(B_2)$.
- (3) An I/O pair of a block B is a pair of signal bunches (i, o) such that $o \in Op(B)(i)$. The set of all I/O pairs of B is denoted as $IO(B)$ and is called the input-output (I/O) relation of B .

An inclusion $o \in Op(B)(i)$ means that o is a possible output of a block B on the input i . For each input i there is some output o . The domain of o is a subset of the domain of i . If o becomes undefined at some time t , but i is still defined at t , we interpret this as an error during the operation of the block B (the block cannot resume its operation after t).

Definition 5. A block B is deterministic, if $Op(B)(i)$ is a singleton set for each $In(B)$ -signal bunch i .

We interpret the operation of a block as a (possibly nondeterministic) choice of an output signal bunch corresponding to a given input signal bunch. However, we would also like to describe this choice as dynamic, i.e. that a block chooses the output signal values at each time t , and in doing so it cannot rely on the future values of the input signals (i.e. values of the input signals at times $t' > t$).

If a block is deterministic, this requirement can be formalized in the same way as the notion of a causal (or nonanticipative) input-output system [26].

Definition 6. A deterministic block B is causal iff for all signal bunches i_1, i_2 and $A \in \mathcal{T}_0$, $o_1 \in Op(B)(i_1)$, $o_2 \in Op(B)(i_2)$, the equality $i_1|_A = i_2|_A$ implies $o_1|_A = o_2|_A$.

This means that the value of the output signal bunch at time t can depend only on the values of the input signal at times $\leq t$.

Some works in the domain of systems theory extend the notion of a causal (deterministic) system to nondeterministic systems. However, there is no unified approach to an extension of this kind. For example, in the work [21], a system, considered as a binary relation on (total) signals $S \subseteq A^T \times B^T$, where T is a time domain (Mesarovic time system, [22]) is “non-anticipatory”, if it is a union of (graphs of) causal (non-anticipatory) selections from S , i.e., $S = \bigcup \{f : \text{dom}(S) \rightarrow \text{range}(S) \mid f \subseteq S, f \text{ is causal}\}$. In the work [23] the authors define another notion of a “non-anticipatory” or “causal” system in nondeterministic case. In the theory developed in the work [24], the authors use a similar notion of a “precausal” system, which is also defined in [22], as a generalization of the notion of a causal system to the nondeterministic case.

In this work, we generalize the notion of a non-anticipatory system in sense of [23] to blocks and call such blocks *nonanticipative*, and generalize the notion of a non-anticipatory system in sense of [21] to blocks, but call such blocks *strongly nonanticipative*. We will show that strongly nonanticipative block is nonanticipative. We will consider the words “causal” and “nonanticipative” as synonyms when they are used informally, but we will distinguish them in the context of formal definitions to avoid a conflict with Definition 6.

Note, however, that the notion of a strongly nonanticipative block defined below is very different from the notion of a “strictly causal” system, defined in some works [25] as a system which uses only past (but not current or future) values of the input signal(s) to produce a current value of the output signal(s).

Definition 7. A block B is nonanticipative, if for each $A \in \mathcal{T}_0$ and $i_1, i_2 \in \text{Sb}(\text{In}(B), W)$, if $i_1|_A = i_2|_A$, then

$$\{o|_A \mid o \in \text{Op}(B)(i_1)\} = \{o|_A \mid o \in \text{Op}(B)(i_2)\}.$$

Definition 8. A block B is a sub-block of a block B' (denoted as $B \trianglelefteq B'$), if $\text{In}(B) = \text{In}(B')$, $\text{Out}(B) = \text{Out}(B')$, and $\text{IO}(B) \subseteq \text{IO}(B')$.

Informally, a sub-block narrows nondeterminism of a block.

Definition 9. A block B is strongly nonanticipative, if for each $(i, o) \in \text{IO}(B)$ there exists a deterministic causal sub-block $B' \trianglelefteq B$ such that $(i, o) \in \text{IO}(B')$.

Informally, the operation of a strongly nonanticipative block B can be interpreted as a two-step process:

1. before receiving the input signals, the block B (nondeterministically) chooses a deterministic causal sub-block $B' \trianglelefteq B$ (response strategy);
2. the block B' receives input signals of B and produces the corresponding output signals (response) which become the output signals of B .

Intuitively, it is clear that in this scheme at any time the block B does not need a knowledge of the future of its input signals in order produce the corresponding output signals.

Lemma 1. *If B is a deterministic block, then B is causal iff B is nonanticipative.*

Proof. Follows immediately from Definition 6.

The following theorem gives a characterization of a nonanticipative block which does not rely on comparison of sets of signal bunches.

Theorem 1. *A block B is nonanticipative iff the following holds:*

- (1) *if $(i, o) \in IO(B)$ and $(i', o') \preceq^2 (i, o)$, then $(i', o') \in IO(B)$;*
- (2) *if $o \in Op(B)(i)$ and $i \preceq i'$, then $(i, o) \preceq^2 (i', o')$ for some $o' \in Op(B)(i')$.*

Proof. (1) Assume that (1) and (2) are satisfied. Assume that $A \in \mathcal{T}_0$, $i_1, i_2 \in Sb(In(B), W)$, and $i_1|_A = i_2|_A$. Let $o \in Op(B)(i_1)$. Then from assumption (1) we have $o|_A \in Op(B)(i_1|_A)$, because $(i_1|_A, o|_A) \preceq^2 (i_1, o)$. Moreover, $i_1|_A \preceq i_2$, because $i_1|_A = i_2|_A$. Thus $(i_1|_A, o|_A) \preceq^2 (i_2, o')$ for some $o' \in Op(B)(i_2)$ by assumption (2). It is not difficult to check that $o|_A \in \{o''|_A \mid o'' \in Op(B)(i_2)\}$. Because i_1, i_2, A are arbitrary, B is nonanticipative by Definition 7.

(2) Assume that B is nonanticipative. Let us prove (1). Assume that $(i, o) \in IO(B)$ and $(i', o') \preceq^2 (i, o)$. Then $i' = i|_A$ and $o' = o|_A$ for some $A \in \mathcal{T}_0$. Then $i'|_A = (i|_A)|_A = i|_A$, whence

$$o' = o|_A \in \{o''|_A \mid o'' \in Op(B)(i)\} = \{o''|_A \mid o'' \in Op(B)(i')\}$$

by Definition 7. Then $o' = o''|_A$ for some $o'' \in Op(B)(i')$. Moreover, $dom(o'') \subseteq dom(i') \subseteq A$. Thus $o' = o''$ and $(i', o') \in IO(B)$.

Let us prove (2). Assume that $o \in Op(B)(i)$ and $i \preceq i'$. Then $i = i'|_A$ for some $A \in \mathcal{T}_0$. Then $i|_A = (i'|_A)|_A = i'|_A$, whence

$$o|_A \in \{o''|_A \mid o'' \in Op(B)(i)\} = \{o''|_A \mid o'' \in Op(B)(i')\}$$

by Definition 7. Then $o|_A = o''|_A$ for some $o'' \in Op(B)(i')$. Moreover, $dom(o) \subseteq dom(i) \subseteq A$, whence $o = o|_A = o''|_A$. Thus $(i, o) \preceq^2 (i', o')$.

Theorem 2. *(About strongly nonanticipative block)*

- (1) *If a block B is strongly nonanticipative, then it is nonanticipative.*
- (2) *There exists a nonanticipative block which is not strongly nonanticipative.*

Proof (Sketch).

(1) Assume that B is strongly nonanticipative. Let \mathcal{R} be the set of all relations $R \subseteq IO(B)$ such that R is an I/O relation of a nonanticipative block. For each $R \in IO$ let us define a block B_R such that $IO(B_R) = R$, $In(B_R) = In(B)$, $Out(B_R) = Out(B)$. Let $\mathcal{B} = \{B_R \mid R \in \mathcal{R}\}$. Then each element of \mathcal{B} is nonanticipative. From Definition 9 and Lemma 1 we have $IO(B) \subseteq \bigcup \mathcal{R} = \bigcup_{B' \in \mathcal{B}} IO(B')$. On the other hand, $IO(B') \subseteq IO(B)$ for any $B' \in \mathcal{R}$, so $IO(B) = \bigcup_{B' \in \mathcal{B}} IO(B')$. It is easy to see from Theorem 1 that (nonempty) union of I/O relations of nonanticipative block is an I/O relation of a nonanticipative block. Thus B is nonanticipative.

(2) Assume that $W = \mathbb{R}$. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be a function that is discontinuous at some point (e.g. a signum function). Let us define a block B such that $In(B) = \{x\}$ and $Out(B) = \{y\}$ for some names x, y , and for each $i \in Sb(In(B), \mathbb{R})$, $Op(B)(i)$ is defined as follows:

- if $dom(i[x]) = T$ and $\lim_{t \rightarrow +\infty} i[x](t)$ exists and finite, then $Op(B)(i)$ is the set of all $\{y\}$ -signal bunches o such that $dom(o) = dom(o[y]) = T$ and

$$\lim_{t \rightarrow +\infty} o[y](t) = f \left(\lim_{t \rightarrow +\infty} i[x](t) \right);$$

- otherwise, $Op(B)(i)$ is the set of all $\{y\}$ -signal bunches o such that $dom(o) = dom(o[y]) = \bigcup \{A \in \mathcal{T}_0 \mid A \subseteq dom(i[x])\}$.

Obviously, in this definition $Op(B)(i) \neq \emptyset$ (because \mathcal{T}_0 is closed under unions) and $dom(o) \subseteq dom(i)$ for each $o \in Op(B)(i)$. So B is indeed a block. Using Theorem 1 it is not difficult to show that B is nonanticipative. Suppose that B has a deterministic causal sub-block B' . Let $a \in \mathbb{R}$ and $a_k \in \mathbb{R}$, $k = 1, 2, \dots$ be a sequence such that $\lim_{k \rightarrow \infty} a_k = a$. Let us show that $\lim_{k \rightarrow \infty} f(a_k) = f(a)$. Let us define sequences $i_k \in Sb(\{y\}, \mathbb{R})$, $o_k \in Sb(\{y\}, \mathbb{R})$, and $t_k \in T$, $k = 1, 2, \dots$ by induction as follows.

Let $i_1(t) = [x \mapsto a_1]$ for all $t \in T$, o_1 be a unique member of $Op(B')(i_1)$, and $t_1 = 0$. If i_1, i_2, \dots, i_k are already defined, let $i_{k+1}(t) = i_k(t)$, if $t \in [0, t_k]$ and $i_{k+1}(t) = [x \mapsto a_{k+1}]$, if $t \in T \setminus [0, t_k]$. Let o_{k+1} be a unique member of $Op(B')(i_{k+1})$. Because $B' \leq B$, $dom(o_{k+1}) = dom(o_{k+1}[y]) = T$ and $\lim_{t \rightarrow +\infty} o_{k+1}[y](t) = f(\lim_{t \rightarrow +\infty} i_{k+1}[x](t)) = f(a_{k+1})$. Then let

$$t_{k+1} = 1 + \max\{t_k, \inf\{\tau \in T \mid$$

$$\sup\{|o_{k+1}[y](t) - f(a_{k+1})| \mid t \geq \tau\} \leq \frac{1}{k+1}\}\}$$

We have defined sequences i_k, o_k, t_k . The sequence t_k , $k = 1, 2, \dots$ is a strictly increasing and unbounded from above and $t_1 = 0$.

Let i be a $\{x\}$ -signal bunch such that $dom(i) = T$, $i(t_1) = i_1(t_1)$, and $i(t) = i_{k+1}(t)$, if $t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}$, and o be a (unique) member of $Op(B')(i)$. We have $i_{k+1}[x](t) = a_{k+1}$ for all $k = 1, 2, \dots$ and $t > t_k$. Then $i[x](t) \in \{a_{k+1}, a_{k+2}, \dots\}$ for all $k \in \mathbb{N}$ and $t > t_k$. For each $\epsilon > 0$ there exists $k \in \mathbb{N}$ such that $|a_{k'} - a| < \epsilon$ for all $k' \geq k$, whence $|i[x](t) - a| < \epsilon$ for all $t > t_k$. Thus $\lim_{t \rightarrow +\infty} i[x](t) = a$. Then $dom(o) = dom(o[y]) = T$ and $\lim_{t \rightarrow +\infty} o[y](t) = f(a)$, because $B' \leq B$.

On the other hand, $i_{k+1}|_{[0, t_k]} = i_k|_{[0, t_k]}$ for all $k \in \mathbb{N}$. Because t_k is an increasing sequence, we have $i_{k'}|_{[0, t_k]} = i_k|_{[0, t_k]}$ for all k and $k' \geq k$. Besides, $i|_{(t_k, t_{k+1}]} = i_{k+1}|_{(t_k, t_{k+1}]}$ for all $k \in \mathbb{N}$, whence $i|_{(t_k, t_{k+1}]} = i_{k'}|_{(t_k, t_{k+1}]}$ for all $k' \geq k + 1$. Also, $i_k(t_1) = i_1(t_1)$ for all $k \in \mathbb{N}$. Then $i|_{[0, t_k]} = i|_{\{t_1\} \cup (t_1, t_2] \cup \dots \cup (t_{k-1}, t_k]} = i_k|_{[0, t_k]}$ for all $k = 2, 3, \dots$, whence $o|_{[0, t_k]} = o_k|_{[0, t_k]}$, because B' is causal. Then $o(t_k) = o_k(t_k)$ for all $k = 2, 3, \dots$, and from the definition of t_k we have $|o[y](t_k) - f(a_k)| = |o_k[y](t_k) - f(a_k)| \leq \frac{1}{k}$ for all $k = 2, 3, \dots$. This implies that $\lim_{k \rightarrow \infty} f(a_k) = f(a)$, because $\lim_{t \rightarrow +\infty} o[y](t) = f(a)$. We conclude that f is sequentially continuous and thus is continuous.

This contradicts our choice of f as a discontinuous function. Thus B has not deterministic causal sub-blocks. Consequently, B is not strongly nonanticipative, though it is nonanticipative.

The proof of this theorem gives a reason of why Definition 9 better captures a intuitive idea of causality than Definition 7. Consider, for example, the block B constructed in the proof of the item (2) of Theorem 2, when f is the signum function (i.e., $f(0) = 0$, $f(x) = 1$, if $x > 0$, and $f(x) < 0$, if $x < 0$). Then B outputs a signal which converges to 1 (as $t \rightarrow +\infty$) whenever the input signal converges to a positive number (as $t \rightarrow +\infty$). Moreover, it outputs a signal which converges to 0 whenever the input signal converges to 0. This implies that when the block receives a decreasing positive input signal which tends to 0, it decides to output values which are close to 0 starting from some time t . Intuitively, after reading the input signal until time t , the block decides that 0 is a more likely limit of the input signal than a positive value, but such a decision cannot be based on the past values of the input signal, so it requires some knowledge of the future of the input signal. These informal observations are captured by the fact that B has no deterministic causal sub-blocks.

In the rest of the paper we will focus on strongly nonanticipative blocks, as more adequate models of (real-time) information processing systems.

Consider an example of a strongly nonanticipative block.

Let u, y be names. Assume that $W = \mathbb{R}$.

Example 1. Let B be a block such that $In(B) = \{u\}$, $Out(B) = \{y\}$, and for each i , $Op(B)(i) = \{o_1(i), o_2(i)\}$, where $o_1(i), o_2(i) \in Sb(Out(B), W)$ are signal bunches such that

- $dom(o_1(i)) = dom(o_2(i)) = dom(i)$;
- $o_1(i)(t) = [y \mapsto i[u](t)]$ for all $t \in dom(i)$;
- $o_2(i)(t) = [y \mapsto 2i[u](t)]$ for all $t \in dom(i)$.

Informally, this means that B is a gain block with a slope which is either 1 or 2 during the whole duration of the block's operation.

Obviously, B satisfies Definition 4(1). Let us check that it is strongly nonanticipative. For $j = 1, 2$ let $B_j \trianglelefteq B$ be a sub-block such that $Op(B_j)(i) = \{o_j(i)\}$ for all $i \in Sb(In(B), W)$ (i.e. B_1 always selects $o_1(i)$ from $Op(B)(i)$ and B_2 always selects $o_2(i)$).

The blocks B_1, B_2 are deterministic and it is easy to see that they are causal. Obviously, each I/O pair $(i, o) \in IO(B)$ belongs either to $IO(B_1)$, or to $IO(B_2)$, so B is strongly nonanticipative.

Now let us consider an example of a block which is not nonanticipative.

Example 2. Let B' be a block such that $In(B') = \{u\}$, $Out(B') = \{y\}$, and

- $Op(B')(i) = \{o_1\}$, where $dom(o_1) = dom(i)$ and $o_1(t) = [y \mapsto 1]$ for all $t \in dom(i)$, if $dom(i[u]) = T$;

- $Op(B')(i) = \{o_2\}$, where $dom(o_2) = dom(i)$ and $o_2(t) = [y \mapsto 0]$ for all $t \in dom(i)$, otherwise.

Informally, the block B' decides whether its input signal u is total. It is easy to see that B' indeed satisfies Definition 4(1), but the condition (1) of Theorem 1 is not satisfied, because $(i, o) \in IO(B')$, where $i(t) = [u \mapsto 0]$ for all $t \in T$, $o(t) = [y \mapsto 1]$ for all $t \in T$, and $(i|_{[0,1]}, o|_{[0,1]}) \preceq^2 (i, o)$, but $(i|_{[0,1]}, o|_{[0,1]}) \notin IO(B')$. So B' is not nonanticipative. Informally, the reason is that at each time t the current value of y depends on the entire input signal.

3.2 Composition of blocks

By connecting inputs and outputs of several (strongly nonanticipative) blocks one can form a larger block – a composition of blocks (see Fig. 2). We assume that an output can be connected to several inputs, but each input can be connected to no more than one output. Unconnected inputs and outputs of constituent blocks become inputs and output of the composition. Connections are interpreted as signal equality constraints and they always relate an output of some block ("source") with an input of the same or another block ("target"). We represent connections in the graphical form (like in Fig. 2) as arrows connecting blocks.

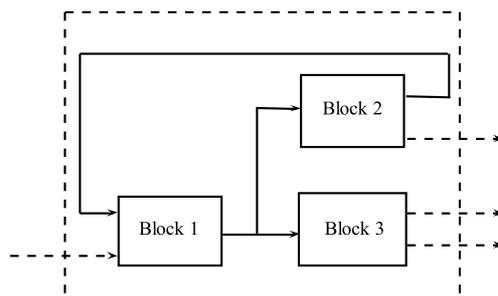


Fig. 2. An informal illustration of a block composition. Three blocks are composed to form a larger block (a dashed rectangle). Solid arrows denote connections between blocks. Dashed arrows denote unconnected inputs/outputs of the blocks 1, 2, 3 (which become the input/outputs of the dashed block).

Definition 10. (1) A block diagram is a pair $((B_j)_{j \in J}, \succ)$ of an indexed family of blocks $(B_j)_{j \in J}$ and an injective binary relation $\succ \subseteq V_{out} \times V_{in}$, which is called an interconnection relation, where

$$V_{in} = \bigcup_{j \in J} \{j\} \times In(B_j), V_{out} = \bigcup_{j \in J} \{j\} \times Out(B_j).$$

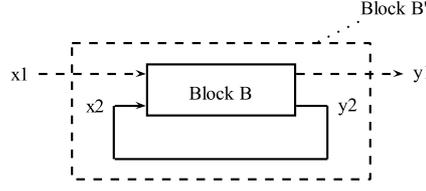


Fig. 3. A strongly nonanticipative block B is composed to obtain a block B' (a dashed rectangle). A solid loop arrow denotes a connection between an output and an input of B . Dashed arrows denote unconnected inputs and outputs of B which become the inputs and outputs of B' .

(2) A block diagram $((B_j)_{j \in J}, \rightsquigarrow)$ is called regular, if each $B_j, j \in J$ is strongly nonanticipative.

Note that a diagram may consist of an infinite set of blocks. A relation $(j, x) \rightsquigarrow (j', x')$ means that the output x of the j -th block is connected to the input x' of the j' -th block.

A block diagram is only a syntactic aspect of a block composition. We define semantics of a block composition only for strongly nonanticipative blocks.

To describe it informally consider Fig. 3. The connection between $y2$ and $x2$ means a signal equality constraint. The block B chooses (nondeterministically) some deterministic sub-block $B_0 \preceq B$. When a signal starts flowing into the input $x1$, the block B' tries to choose the initial values for the signals of $y1, y2, x2$ so that they satisfy the operation of the block B_0 ($Op(B_0)$) and the signals of $x2$ and $y2$ have the same values. If such initial values do not exist, the block B' terminates (the output signal bunch is nowhere defined). Otherwise, B' continues to operate in the similar way until either the signals of $y1, y2, x2$ cannot be continued, or the input signal ($x1$) ends.

Definition 11. Let $((B_j)_{j \in J}, \rightsquigarrow)$ be a regular block diagram.

A block B is a composition of $(B_j)_{j \in J}$ under the interconnection relation \rightsquigarrow , if

- $In(B) = (\bigcup_{j \in J} \{j\} \times In(B_j)) \setminus range(\rightsquigarrow)$,
- $Out(B) = (\bigcup_{j \in J} \{j\} \times Out(B_j)) \setminus dom(\rightsquigarrow)$,
- $Op(B)(i)$ is the set of all $Out(B)$ -signal bunches o such that there exist deterministic causal sub-blocks $B'_j \preceq B_j, j \in J$ and an indexed family $(i_j, o_j)_{j \in J} \in X_m(i)$ such that
 - (1) $dom(o) = dom(o_j)$ for all $j \in J$,
 - (2) $o[(j, x)] = o_j[x]$ for all $(j, x) \in Out(B)$,
 where $X_m(i)$ is the set of $\preceq^{J,2}$ -maximal elements of $X(i)$, and $X(i)$ is the set of all indexed families of pairs of signal bunches $u = (i_j, o_j)_{j \in J}$ such that
 - (3) $dom(i_j) = dom(o_j) = dom(i_{j'}) = dom(o_{j'}) \subseteq dom(i)$ for all $j, j' \in J$,
 - (4) $i_j[x] = i|_{dom(i_j)}[(j, x)]$ for each $(j, x) \in In(B)$,
 - (5) $(i_j, o_j) \in IO(B'_j)$ for each $j \in J$,

(6) $(j, x) \mapsto (j', x')$ implies $o_j[x] = i_{j'}[x']$.

In this definition, i_j and o_j denote the input and output signal bunches of the j -th block. The set $X_m(i)$ contains maximally extended (in sense of the relation $\preceq^{J,2}$) indexed families of signal bunches defined on a subset of the domain of i (the input signal bunch of B) which satisfy constraints imposed by the interconnection relation. Any such family gives a possible output of B for the given i by the condition (2), i.e. output signals of B are obtained from the output signals of the sub-blocks B'_j .

It is clear that any two compositions of $(B_j)_{j \in J}$ under \mapsto are semantically identical.

Lemma 2. *(Continuity of the operation of a causal deterministic causal block). Let B be a deterministic causal block. Let $c \subseteq \text{Sb}(\text{In}(B), W)$ be a non-empty \preceq -chain, i^* be its supremum (in sense of \preceq), and $o^* \in \text{Op}(B)(i^*)$. Then o^* is a supremum of $\bigcup_{i \in c} \text{Op}(B)(i)$ (in sense of \preceq).*

Proof. Follows from Definition 6.

Theorem 3. *Let $((B_j)_{j \in J}, \mapsto)$ be a regular block diagram. Then*

- (1) *A composition of $(B_j)_{j \in J}$ under \mapsto exists.*
- (2) *If B is a composition of $(B_j)_{j \in J}$ under \mapsto , then B is strongly nonanticipative.*

The proof follows from Lemma 2 and Definition 11.

3.3 Specification and implementation

Above we have considered a block as an abstract model of a real component. However, it can also be considered as a specification of requirements for a component. Let B^{spec}, B^{impl} be two strongly nonanticipative blocks. Let us call them a specification block and implementation block respectively.

Definition 12. B^{impl} is a refinement of B^{spec} , if B^{impl} is a sub-block of B^{spec} .

I.e. an implementation should have the the same input and output names as a specification, and for each input, an output of an implementation should be one of the possible outputs of a specification. We generalize this to diagrams.

Let $D = ((B_j)_{j \in J}, \mapsto)$ and $D' = ((B'_j)_{j \in J'}, \mapsto')$ be regular block diagrams.

Definition 13. D is a refinement of D' , if $J = J'$, B_j is a refinement of B'_j for each $j \in J$, and the relations \mapsto and \mapsto' coincide.

Theorem 4. *(Compositional refinement) Let B be a composition of $(B_j)_{j \in J}$ under \mapsto and B' be a composition of $(B'_j)_{j \in J'}$ under \mapsto' . If D is a refinement of D' , then B is a refinement of B' .*

The proof follows from Definition 9, 11, and transitivity of the sub-block relation.

This theorem can be considered as a foundation of a modular approach [29, 30] to system design:

1. Create specifications of the system components $(B'_j, j \in J')$ and connect them (\mapsto') , as if they were real components.
2. Analyze a composition of specifications (B') to ensure that any of its implementations $(B'' \trianglelefteq B')$ satisfies requirements to the final system.
3. Create an implementation (B_j) for each specification (B'_j) .
4. Connect implementations (according to \mapsto'). Then the composition of implementations (B) is a final system which satisfies design requirements.

We consider the steps 1 and 3 domain- and application-specific. The conclusion of the step 4 is addressed in Theorem 4. Step 2 requires some verification method which depends on the nature of requirements.

One of the most basic and common requirements is that the operation of B' is defined on all input signal bunches which are possible in the context of a specific application of this composition. This trivially holds because of Theorem 3 and our definition of a block. However, B' may terminate abnormally on some or all input signal bunches of interest (as we have noted, we interpret the situation when $o \in Op(\tilde{B})(i)$ and $dom(o) \subset dom(i)$ for some block \tilde{B} as abnormal termination of \tilde{B} on i). So the requirement can be reformulated as follows: B' never terminates abnormally on any input signal bunch from a given set IN (this implies that the same property holds for B). We will call this property as well-definedness of the operation of B' on IN and study it in the next subsection.

3.4 Well-definedness of the operation of a composition of blocks

Let B be a block and IN be some set of $In(B)$ -signal bunches.

Definition 14. *The operation of B is well-defined on IN , if $dom(i) = dom(o)$ for each $i \in IN$ and $o \in Op(B)(i)$.*

Let $D = ((B_j)_{j \in J}, \mapsto)$ be a regular block diagram and B be a composition of $(B_j)_{j \in J}$ under \mapsto . Let \mathcal{F} be the set of all families of blocks of the form $(B'_j)_{j \in J}$, where for each $j \in J$, B'_j is a deterministic causal sub-block of B_j .

For each $In(B)$ -signal bunch i and a family of blocks $F = (B'_j)_{j \in J}$ let $X^F(i)$ be the set of all indexed families of pairs of signal bunches $u = (i_j, o_j)_{j \in J}$ which satisfy conditions (3)-(6) of Definition 11 (for B and $(B'_j)_{j \in J}$).

Let $X_m^F(i)$ denote the set of all $\preceq^{J,2}$ -maximal elements of $X^F(i)$. For any indexed family of signal bunches $u = (i_j, o_j)_{j \in J}$ let $O(u)$ denote the set of all $Out(B)$ -signal bunches o which satisfy conditions (1)-(2) of Definition 11.

For any $u = (i_j, o_j)_{j \in J} \in X^F(i)$, the domains of i_j, o_j for all $j \in J$ coincide. Denote by $cdom(u)$ this common domain (we assume $cdom(u) = T$, if $J = \emptyset$).

From Definition 11 we have $Op(B)(i) = \bigcup_{F \in \mathcal{F}} \bigcup_{u \in X_m^F(i)} O(u)$ for each i . Then from Definition 14 we get the following simple criterion:

Theorem 5. *The operation of B is well-defined on IN iff for each $i \in IN$, $F \in \mathcal{F}$, and $u \in X^F(i)$, if $cdom(u) \subset dom(i)$, then $u \notin X_m^F(i)$.*

This criterion means that B is well-defined, if each $u \in X^F(i)$, the common domain of which does not cover $\text{dom}(i)$, is extendable to a larger $u' \in X^F(i)$ (in sense of $\preceq^{J,2}$). We will call it a local extensibility criterion, because, basically, to prove well-definedness, we only need to show that the members of the family u can be continued onto a time segment $[0, \sup \text{cdom}(u) + \epsilon]$ for some small $\epsilon > 0$ (under constraints imposed by the interconnection relation \rightarrow). Locality is especially useful when a block diagram contains "delay" blocks (possibly working as variable delays), because constraints imposed by connections between blocks reduce over small time intervals.

A drawback of this criterion is that it requires checking local extensibility of signal bunches satisfying the I/O relations ($IO(B'_j)$) of arbitrarily chosen deterministic causal sub-blocks $B'_j \preceq B_j$ (condition (5) of Definition 11), which are not explicitly expressed in terms of I/O relations of B_j , $j \in J$.

For this reason, we seek for a condition of well-definedness in terms of I/O relations of the constituents of the composition ($IO(B_j), j \in J$).

Let $X(i)$ denote the set $X^F(i)$, where $F = (B_j)_{j \in J}$. Note that F may not be a member of \mathcal{F} .

Theorem 6. *The operation of B is well-defined on IN iff for each $i \in IN$ and $u \in X(i)$ there exists $u' \in X(i)$ such that $u \preceq^{J,2} u'$ and $\text{cdom}(u') = \text{dom}(i)$.*

The proof follows from Theorem 5 and Definition 9.

References

1. Simulink - Simulation and Model-Based Design, <http://www.mathworks.com/products/simulink>
2. Campbell, S.L., Chancelier, J.-P., Nikoukhah, R.: Modeling and Simulation in Scilab/Scicos with ScicosLab 4.4. Springer (2010)
3. Multi-Engineering Modeling and Simulation – Dymola, <http://www.3ds.com/products/catia/portfolio/dymola>
4. SCADE Suite, <http://www.esterel-technologies.com/products/scade-suite>
5. Caspi, P., Pilaud, D., Halbwegs, N., Plaice, J.: LUSTRE: A declarative language for programming synchronous systems. In: 14th Annual ACM Symp. on Principles of Programming Languages, Munich, Germany, pp. 178-188 (1987)
6. Henzinger, T., Horowitz, B., Kirsch, C.: Giotto: A Time-Triggered Language for Embedded Programming. First International Workshop on Embedded Software, EMSOFT'01, pp. 166-184 (2001)
7. Lubliner, R., Tripakis, S.: Modular Code Generation from Triggered and Timed Block Diagrams. In: IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 147-158 (2008)
8. Sontag, E.D.: Mathematical Control Theory: Deterministic Finite Dimensional Systems. Second Edition, Springer, New York (1998)
9. Proakis, J., Manolakis, D.: Digital Signal Processing: Principles, Algorithms and Applications, 4th ed. Pearson (2006)
10. Tiwari, A.: Formal semantics and analysis methods for Simulink Stateflow models. Unpublished report, SRI International (2002)

11. Bouissou, O., Chapoutot, A.: An operational semantics for Simulink's simulation engine. *LCTES 2012*, pp. 129-138. (2012)
12. Agrawal, A., Simon, G., Karsai, G.: Semantic translation of Simulink/Stateflow models to hybrid automata using graph transformations. *Electronic Notes in Theoretical Computer Science* 109, 43-56 (2004)
13. Marian, N., Ma, Y.: Translation of Simulink Models to Component-based Software Models. In: 8-th Int. Workshop on Research and Education in Mechatronics, 14-15 June 2007, Talin University of Technology, Estonia (2007)
14. Pinto, R., Sangiovanni-Vincentelli, A., Carloni L.P., Passerone, R.: Interchange formats for hybrid systems: Review and proposal. In: *HSCC 05: Hybrid Systems Computation and Control*. Springer-Verlag, pp. 526-541 (2005)
15. Beek, D.A., Reniers, M.A., Schiffelers, R.R., Rooda, J. E.: Foundations of a Compositional Interchange Format for Hybrid Systems. In: *HSCC'07*, pp. 587-600 (2007)
16. Henzinger, T.: The theory of hybrid automata. In: *IEEE Symposium on Logic in Computer Science*, pp. 278-292 (1996)
17. Goebel, R., Sanfelice, R., Teel, R.: Hybrid dynamical systems. In: *IEEE Control Systems Magazine* 29, 29-93 (2009)
18. Schrammel, P., Jeannot, B.: From hybrid data-flow languages to hybrid automata: a complete translation. In: *HSCC 2012*: pp. 167-176 (2012)
19. Camhbel, M.,K., Heemels, A.J., van der Schaft, A.J., Schumacher, J.M.: Solution concepts for hybrid dynamical systems. In: *Proc. IFAC 15th Triennial World Congress*, Barcelona, Spain (2002)
20. Nikitchenko, N.S.: A composition nominative approach to program semantics. Technical report IT-TR 1998-020, Technical University of Denmark, 103 p. (1998)
21. Windeknecht, T.G.: Mathematical systems theory: Causality. *Mathematical systems theory* 1, pp. 279-288 (1967)
22. Mesarovic, M.,D., Takahara, Y.: *Abstract systems theory*. Springer, Berlin Heidelberg New York, 439 p. (1989)
23. Foo, N., Peppas, P.: Realization for Causal Nondeterministic Input-Output Systems. *Studia Logica* 67, pp. 419-437 (2001)
24. Lin, Y.: *General systems theory: A mathematical approach*. Springer, 382 p. (1999)
25. Matsikoudis, E., Lee, E.: On Fixed Points of Strictly Causal Functions. Technical report UCB/EECS-2013-27, EECS Department, University of California, Berkeley (2013).
26. Williams, J.: Paradigms and puzzles in the theory of dynamical systems. In: *IEEE Transactions on Automatic Control* 36, pp. 259-294 (1991)
27. Williams, J.: On Interconnections, Control, and Feedback. In: *IEEE Transactions on Automatic Control* 42, pp. 326-339 (1997)
28. Williams, J.: The behavioral approach to open and interconnected systems. In: *IEEE Control Systems Magazine*, pp. 46-99 (2007)
29. Baldwin, C.Y., Clark, K.B.: *Design Rules, Volume 1: The Power of Modularity*. MIT Press (2000)
30. Tripakis, S., Lickly, B., Henzinger, T., Lee, E.: On relational interfaces. *Proceedings of EMSOFT'2009*, pp. 67-76 (2009)
31. Ivanov, Ie.: A criterion for global-in-time existence of trajectories of non-deterministic Markovian systems. *Communications in Computer and Information Science* 347, pp. 111-130, Springer (2012)
32. Carloni, L.P., Passerone, R., Pinto A.: Languages and Tools for Hybrid Systems Design. *Foundations and Trends in Design Automation* 1, 1-204 (2006)

Multilevel Environments in Insertion Modeling System

Dmitriy M. Klionov¹

¹Kherson State University, 40 rokiv Zhovtnya st. 27, Kherson, Ukraine

soulslayermaster@gmail.com

Abstract. The goal of this paper is to show that the Insertion Modeling System[1] developed by A.A. Letichevsky of the department 100/105 of the Glushkov Institute of Cybernetics, National Academy of Science of Ukraine, Kyiv, Ukraine, can be used as an instrument for the modeling and analysis of complex distributed systems, such as a client-server architectures. The Insertion Modeling [1] is based on the interactions of environments and agents inserted into that environments. Agents have different behaviors represented as Behavior Algebras, and can also be the environments themselves, having another agents with different behaviors inserted into them. The definition for multilevel environments was first given in a paper [1], and was slightly extended in following papers.

Keywords. Insertion modeling, multilevel environments, compatibility relation, client-server architecture

Key terms. Computation, Model, Insertion Modeling

1 Introduction

Insertion modeling is a technology for specification and verification of complex distributed systems based on the interactions of agents and environments. Agents and environments are models of some entities of real world or components of complex systems on different levels of abstraction that interact with one another by means of insertion functions. Also if the environment is considered as an agent it can also be inserted to other environments. In order to model complex systems those consist of a lot of components that have hierarchical structure, the notion of multilevel environments, with agents that are able to move from one environment to another is required. The notion of mobility of such mobile agents are based on the approach recently favored in declarative mobile language design is using mobile calculi that extend or modify the π -calculus [10] with new features, including mechanisms for encryption and security. Calculi of this kind include, among others, the Spi Calculus [6], and the Ambient Calculus [7]. In addition, there is a broader body of work favoring declarative approaches, including work in the field of coordination languages. There has also

been a great expansion of the capabilities and security of agent-based languages such as OAA [10] and D'Agents[13].

According to the Ambient Calculus [7], devised by Luca Cardelli the main difficulty of mobile computations in Web is not in mobility itself but in handling of administrative domains. In the early days of the Internet one could rely on a flat name space given by IP addresses; knowing the IP address of a computer would very likely allow one to talk to that computer in some way. This is no longer the case: firewalls partition the Internet into administrative domains that are isolated from each other except for rigidly controlled pathways. System administrators enforce policies about what can move through firewalls and how.

The client-server model is the prevalent approach in computer networking. The model assigns one of two roles to the computers in a network: a client or a server. A server is a computer system that selectively shares its resources; a client is a computer or computer program that initiates contact with a server in order to make use of a resource. Data, CPUs, printers, and data storage devices are some examples of resources. This model can be represented as a set of administrative domains, with defined access rules, or as some architectural design pattern, like three-tier pattern. Both of these are presented in this paper in terms of the insertion modeling.

2 Insertion Modeling System

Insertion modeling system is an environment for the development of insertion machines and performing experiments with them. Insertion model of a system represent this system as a composition of environment and agents inserted into it, using the insertion function. Contrariwise the whole system as an agent can be inserted into another environment. In this case we speak about the internal and external environment of a system. Agents inserted into the internal environment of a system themselves can be environments with respect to their internal agents. In this case we speak about multilevel structure of agent or environment and about high level and low level environments.

Agent and environments have a set of action and a set of behaviors (processes), defined in behavior algebra. Two set of actions: a set of environment actions and a set of agent actions define the type of environment. If an agent is about to be inserted into the environment at least one of its actions must be allowed by this environment. So the set of agent actions define the type of environments it can be inserted in, as well as the environment's set of allowed agent actions define the type of agents that can be inserted into this environment. Such a relation between types of agents and environments is called compatibility relation [2], which defines a directed graph. When an agent is inserted into some environment, it is able to move to another environment if it is compatible with this environment. For example the rule(1) shows an agent u that moves to an external environment E , from environment R , it is currently inserted into.

$$\frac{u \xrightarrow{\text{move } e} u'}{E[R[u]] \xrightarrow{\text{move_up}(r \rightarrow e)} E[u', R[]]} P(E, R, u, \text{move } e) \quad (1)$$

Here e and r – are the names of environments, $R[]$ – describes environment R that currently have no agents inserted into it. Insertion only occurs if a predicate P is true, and in general case it may depend only on the types of agents and environments. This example rule shows “one step” movement of an agent u , and if the new state of agent u' has the same type as u , and types of environments E and R had not changed as well, rule (1) can be considered as commutative. Also “long range” movements can be defined recursively, for any set of environments between E and R .

3 Insertion Models of Client-Server Architecture

3.1 Domain Model

This model describes a client-server-architecture as a set of administrative domains that have certain access rules. Each of these domains is represented by an environment in IMS. Agents are messages that travel over these domains, trying to access certain protected area of some administrative domain of the server. As an example we take our website `apsystem.or.ua`. It is shown at picture below. The top-most environment E - represents some network (local-area network or internet), with environments of `apsystem` itself, and a set of clients $C1, C2, \dots, Cn$ inserted into the network. Client environments create agents and send them over the network in order to gain access to some function of `apsystem` if they have certain permission, or to a domain of another client. One of the clients can represent a villain (Hacker), which goal is to find all possible security risks and ways of an attack to curtain security protocol.

In order to access administrative domain and to authorize on a server the client has to show that it knows some secret, which is only known to client and server (or two clients that want to exchange some data), and which is not transferred over the network. This key is used to encode messages (transferred by the agents), and when agents tries to move into the environment of administrative domain, this key is used to decode the message, if it is possible than agent inserts into the environment, and proceeds further. There are many ways for generating such secret.

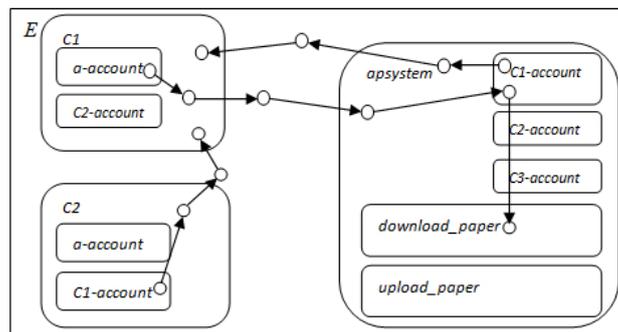


Fig. 1. The domain model of client server

This model uses the standard Needham–Schroeder Public Key protocol [21]. Each client and server has a secret key, which is used to decode messages encoded with appropriate public key. When an agent gets inside the administrative domain (apsystem for example), it have to get a permissions to act inside it. The message transferred by this agent, contains the information about the access rights of the client who sent it. This data is used to move further. When an agent reaches some function (“download_paper” for example) it has a permission to, it is to be sent back by the server to the client. Account environments that are inserted into the clients and the top-most environment of the server store all information required to authorize at appropriate client or server. Tables below show all types of environments and agents.

Types of environments of clients and the top-most environment of the server, are identical. In general the client differs from the server only by the means of environments inside it, which require an action authorized_move.

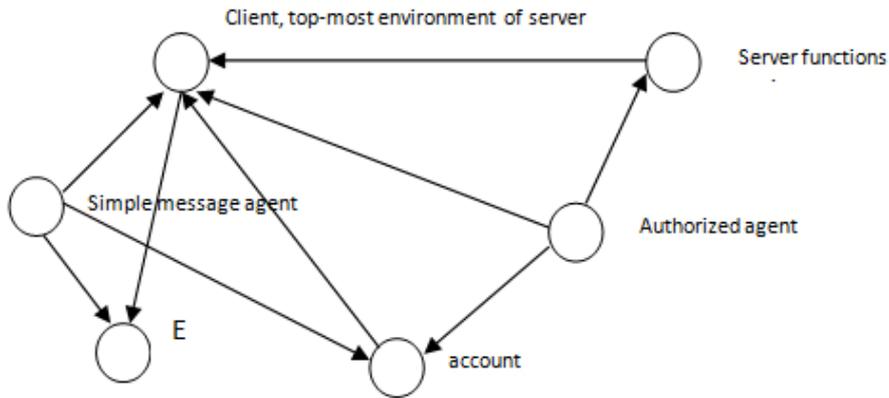


Fig. 2. Compatibility graph for the client-server domain model

Vertexes represent agents and environments, and edges represent a compatibility relation. Directions mean that for example the authorized agent can be inserted into the environments of the account, server functions environments, clients and servers environments.

Interactions with agents:

$$\frac{u \xrightarrow{\text{send } a} u'}{E[C[AC[u]], A[]] \xrightarrow{\text{send } a} E'[C'[AC'[]], A'[u']]} P(E, C, AC, A, u, \text{send } a) \quad (2)$$

In equation (2) send(A) Means that client C sends the message u, with an appropriate account AC, to the server A[], over the network E, where a – is the name of server A[]. The definition A[] means that there were no agents inserted into this environment.

Table 1. Actions of agents and environments in domain model

Agent / Environments type	Attributes	Actions
Simple message	<i>mb</i> – message body, actual information carried by this agent;	<i>send a</i> - makes agent to move to the server environment named <i>a</i> ,
	<i>sender</i> – the name of the one who sent this message;	
	<i>enc key</i> – key that is used by encryption algorithm;	<i>access d</i> - agent tries to authorize in order to enter the environment named <i>d</i> , that is in the server environment.
Authorized message	<i>mb</i> – message body, actual information carried by this agent;	<i>auth move d</i> - “authorized move” to some internal environments of the server named <i>d</i>
	<i>role</i> – defines the access level of this information	<i>get_data(x)</i> - agent shares the data it carries.
		<i>invoke(x)</i> - invokes the main function of the environments of the server functions, <i>x</i> – is the access level of authorized agent. It receives as an answer or the result of execution of function, or the “access denied” message.
<i>done(x)</i> - required to check if the result it carries is equivalent to the expected result		
Clients and the top-most environment of the server Allowed actions: send, access, authmove	Secretkey – an integer value of the client’s secret key, that is used by the Needham-Schroeder algorithm Nounce – a place for random numbers.	<i>allow(y)</i> - environment checks the incoming message from the server, <i>y</i> – is the secret key that is used to decode the information from that message.
Accounts Allowed actions: access, authmove, send, get_data(y), done(z)	<i>server</i> – the name of the server it belongs to	<i>update(x)</i> - account is able to update its data about the secret keys used in the Needham-Schroeder algorithm
	<i>role</i> – an integer value that represent a role of this account at server	<i>check_goal(x)</i> - checks if the result brought by the message, is equal to the expected result that is <i>x</i>
	<i>publickey</i> – the public key of the server, that is used by the Needham-Schroeder algorithm	<i>create(r,t)</i> - environment creates agent named <i>r</i> , which has type <i>t</i>
	<i>secret</i> – that will be obtained by Needham-Schroeder algorithm	
Environments that represent server functions(download_paper, upload_paper) Allowed actions: authmove, invoke	<i>permission</i> – an integer value indicating what the required permissions to access it are.	<i>check_permission u</i> - checks if the access level of agent <i>u</i> is appropriate for performing action, if it do then it is delta, if not then agent receives a message that it has no rights to perform the function of this environment
E Allowed actions: send a		

$$\frac{A \xrightarrow{\text{allow}(y)} A', u \xrightarrow{\text{access } ca} u'}{A[u, CA[], D[]] \xrightarrow{\text{access } ca} A[CA[u], D[]]} P(A, CA, u, \text{access } ca) \quad ; \quad (3)$$

An agent u tries to gain access to the server $A[]$, A tries to authorize it, using the secret y , if the authorization succeeds, then u enters appropriate account on the server that is CA , and ca is its name.

$$\frac{CA \xrightarrow{\text{create}(r,t)} CA', u \xrightarrow{\text{get_data}(x)} u'}{A[CA[u], D[]] \xrightarrow{\text{create}(r,t) \text{ } ca} A[CA[u', r], D[]]} P(CA, u, t, \text{get_data}(x), \text{create}(r, t)) \quad (4)$$

An account environment CA creates a new agent named r , which type is t . It carries all data received from u , by the action $\text{get_data}(x)$, x – is that data. This rule creates an agent of type `authorized_agent`, but it can create an agent of any type that is compatible with this environment.

$$\frac{r \xrightarrow{\text{authmove } d} r'}{A[CA[u, r], D[]] \xrightarrow{\text{authmove } d} A'[CA'[u], D'[r']] } P(A, CA, D, r, \text{authmove } d) \quad (5)$$

The authorized agent u moves to the environment $D[]$, that represent one of the server functions;

$$\frac{D \xrightarrow{\text{check_permission } u} D', r \xrightarrow{\text{invoke}} r'}{D[r] \xrightarrow{\text{invoke}} D'[r']} P(D, r, \text{check_permission } r, \text{invoke}) \quad (6)$$

The agent u invokes the main function of $D[]$, and depending on the result of `check_permission` u , the result of this `invoke` might be different.

$$\frac{AC \xrightarrow{\text{check_goal}(x)} AC' r \xrightarrow{\text{done}(y)} \Delta}{AC[r] \longrightarrow AC[\Delta]} P(AC, r, \text{check_goal}(x), \text{done}(y)) \quad (7)$$

When an agent comes back to the client that sent it, the client checks the message it carried, and it matches the required result then it is successful termination. These rules only work if both the client and the server share a secret, known only to them. In order to safely generate such secret the Needham–Schroeder public key algorithm is used. Usually the Needham–Schroeder protocol requires a second server that hosts all the public keys, but for simplicity we assume that all clients and servers know all the public keys. If the secret has already been created, than it is taken instead of public key and secret key for encoding and decoding of messages.

It runs as follows:

1. First we check if the *secret* exists for an account A , if not we send message to the server $A[]$ by the rule (2), and set the value of an agent's attribute *mb* to N_i that is a simple random number.
2. Then the server $A[]$ uses the rule(3) to decode message using the secret key of server $A[]$,.if the *secret* is not created yet.

3. Then the server replies by the rule (2) to client C the value of mb is set to (N_1, N_2) , N_1 – is the random number created by the client C, and N_2 – is the new random number.
4. If the first part of the mb is equal to the random number that was generated before, than C can take the pair (N_1, N_2) , as a *secret* for the account A.
5. Then C sends a message to A, that contains N_2 . When A will receive it, he will also take the pair (N_1, N_2) , as a *secret* for account C.

In order to verify this protocol one of the clients has to take the role of a villain, its goal is to be authorized as another client from the network, using in this case a man-in-middle attack. [22]

3.2 Insertion Model of Three-Tier Architecture

Unlike of the previous model this one focuses on the actual behavior of data-packages represented by agents, inside the server environment, divided basically to three layers according to the three-tier architecture. The example model of the server hosting two sites apsystem and unarea, is presented.

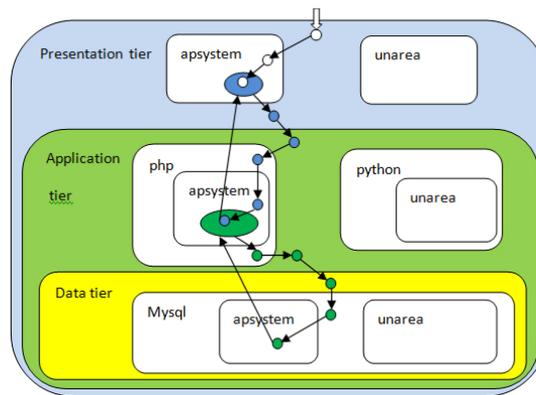


Fig. 3. Insertion model of three-tier client-server architecture

Their frontends are located inside the presentation tier.

$$E[\text{Pr}[\text{aps} [], \text{un} [], [\text{App}[\text{Php}[\text{aps_l}[]], \text{Py}[\text{un_l}[]], [\text{Data}[\text{mysql}[\text{aps_d}[]], \text{un_d}[]]]]]] \tag{8}$$

(8) is the state of environment in such example. E – the top-most environment, Pr – the presentation tier, aps – the apsystem frontend, un – the unarea frontend, App – the application tier, PHP\ PY – all sites developed in php and python accordingly, aps_l\un_l – the logic of apsystem\unarea, Data – the data tier, aps_d\un_d – the data-base of apsystem\unarea. The user only works with frontend. This means, that the

incoming agent is compatible only with environments of the presentation layer. An agent inserted into one of the frontends carries one request.

Table 2. Types of agents and environments

Agents/Environments types	Actions
User request	execute(x) - executes the request brought by user, x-is the request data User_move d - User agent moves to environment, named d
Script request Allowed actions: execute(x), User_move d	execute_script(y) - executes the request brought by script, y-is the request data Script_move d - script agent moves to environment named d
Data base request Allowed actions: execute_script(y), Script_move d	Execute_query(z) - Executes the request brought by data base, z-is the request data Data_base_move d - Data base agent moves to environment named d
Environments of the Presentation tier Allowed actions: execute(x), User_move d, Script_move d	Create (r,t) - Creates agent named r, of the type t
Environments of the Application tier Allowed actions: execute_script(x), Script_move d, Data_base_move d	Create (r,t) - Creates agent named r, of the type t
Environments of the Data tier Allowed actions: execute_query(x), Script_move d, Data_base_move d	

Interaction with environments: In the rule (9) u gets inside Pr using user_move pr, where pr is the name of the environment Pr, if P can allow this. (a simple one step insertion). The same way u gets inside the environment aps[], using in(aps). This shows how a user goes to some web-site (apsystem in our case), in order to download a page for example. In order to do so he has to load a web-page that has a required link to the paper he wants.

$$\frac{u \xrightarrow{\text{user_move } pr} u'}{E[u, Pr[APS], App[APS_l], Dat[mysq[APS_d]]] \xrightarrow{\text{user_move } pr} P(Pr, u, \text{user_move } pr)} \quad (9)$$

$$E[Pr[u', APS], App[APS_l], Dat[mysq[APS_d]]]$$

The link to the paper is stored in the site's data base that is inside the Data tier, and the rules for extracting these data, and displaying them is inside the Application layer. So, the frontend part (environment of apsystem in our case) creates a new agent, that is compatible only with this environment, and with according environments of the Application layer:

$$\frac{u \xrightarrow{\text{execut}(x)} \Delta}{E[\text{Pr}[APS[u], Q]] \xrightarrow{\text{execut}(x)} E'[\text{Pr}'[APS'[\Delta], Q]]} P(APS, u, \text{execut}(x)) \quad (10)$$

Here we check if u is able to execute its request x if it succeeds than it is DELTA, if it is NOT able to, than we have to check if there any environments inside aps , that are compatible with u , go inside them, and again try the same rule. Q is put for simplicity; it describes all rest of environments that are not involved in the current rule. If there are no such environments or even after insertion to such environment u is still unable to solve(x), then we have to create a new agent r that will get necessary data from application tier.

$$\frac{APS \xrightarrow{\text{create}(r,t)} APS'}{E[\text{Pr}[aps[u], Q]] \xrightarrow{\text{create}(r,t)} E'[\text{Pr}'[aps'[u, r], Q]]} (APS, t, \text{create}(r, t)) \quad (11)$$

Note: r has to be created inside that environment, which u is currently inserted in.

$$\frac{r \xrightarrow{\text{execute_script}(y)} r'}{E[\text{Pr}[APS[u], App[PHP[APS_l[r]]]]] \xrightarrow{\text{solve}(y)} E'[\text{Pr}'[APS[u], App'[PHP[APS_l[r']]]]]} P(APS_l, r, \text{execute_script}(y)) \quad (12)$$

The agent r moves to that environment (APS_l). It should go first to the environment PHP, which is the top-environment of all sites based on PHP, and then moves to the environment APS_l . The rule for its movement is similar to the movement of a user request. The execution of script request is different:

$$\frac{r \xrightarrow{\text{script_move_aps}} r'}{E[\text{Pr}[APS[u], App[PHP[APS_l[r]]]]] \xrightarrow{\text{script_move_aps}} E'[\text{Pr}'[APS[u, r'], App'[PHP[APS_l[r']]]]]} P(PR, App, PHP, APS, APS_l, r, \text{script_move_aps}) \quad (13)$$

If the execution succeeds, than r moves back to the environment, which created it. If not, then the environment APS_l , creates a new agent, which is the query for the data tier. The rules are similar.

4 Conclusions

The client-server model can be considered as a prevalent approach in computer networking, and is one of the best examples of complex distributed systems. Two examples of insertion models of client-server architecture are presented in this paper: the domain model – as a set of administrative domains with pre-defined access rules; and a three-tier architecture - a client-server architecture in which the presentation, the application processing, and the data management functions are logically separated. Both these insertion models with multilevel environments and mobile agents can be extended later for more complicated applications, such as the verification of crypto-

graphic protocols, the problem solving, the constraint propagation, the cognitive architectures.

References

1. Letichevsky, A. A.: Insertion Modeling. *Control Systems and Computers*, 6, 3-14 (2012)
2. Letichevsky, A.: Algebra of Behavior Transformations and its Applications. In: Kudryavtsev, V. B., Rosenberg, I. G. (eds.) *Structural Theory of Automata, Semigroups, and Universal Algebra*. NATO Science Series II. Mathematics, Physics and Chemistry, vol. 207, pp. 241-272, Springer (2005)
3. Baranov, S., Jervis, C., Kotlyarov, V., Letichevsky, A., Weigert, T.: Leveraging UML to Deliver Correct Telecom Applications. In: L. Lavagno, G. Martin, and B. Selic (eds.) *UML for Real: Design of Embedded Real-Time Systems*. Kluwer Academic Publishers, Amsterdam (2003)
4. Letichevsky, A.A., Kapitonova, J., Letichevsky, A. Jr., Volkov, V., Baranov, S., Kotlyarov, V., Weigert, T.: Basic Protocols, Message Sequence Charts, and the Verification of Requirements Specifications. *Computer Networks*, (47), 662–675 (2005)
5. Kapitonova, J., Letichevsky, A., Volkov, V., Weigert, T.: Validation of Embedded Systems. In: R. Zurawski (ed.) *The Embedded Systems Handbook*. CRC Press, Miami (2005)
6. Abadi, M., Gordon, A.D.: A Calculus for Cryptographic Protocols: the spi Calculus. In: *Proc. 4th ACM Conference on Computer and Communications Security*, pp. 36-47, (1997)
7. Cardelli, L., Gordon, A.: Mobile Ambient. In: Nivat, M. (ed.) *Proc. FoSSaCs'98: Foundations of Software Science and Computational Structures*, LNCS 1378, pp. 140–155, Springer-Verlag (1999)
8. Lange, D.B., Oshima: *Programming and Deploying Java Mobile Agents with Aglets*. Addison-Wesley (1998)
9. Kotz, D., Gray, R.S.: Mobile Agents and the Future of the Internet. *ACM Operating Systems Review* 33(3), 7–13, (1999)
10. Milner, R., Parrow, J., Walker, D.: A Calculus of Mobile Processes (Parts I and II). *Information and Computation* 100, pp. 1-77 (1992)
11. Martin, D., Cheyer, A., Morgan, D.: The Open Agent Architecture: A Framework for Building Distributed Software Systems. *Applied Artificial Intelligence*, 12, 91–128 (1999)
12. Gray, R.S., Kotz, D., Cybenko, G., Rus, D.: D'Agents: Security in a Multiple-Language, Mobile-Agents System. In: Vigna G. (ed.) *Mobile Agents and Security*, LNCS 1419, pp. 154–187, Springer-Verlag (1998)
13. Milner, R.: *A Calculus of Communicating Systems*, LNCS 92, Springer-Verlag. (1980)
14. Milner, R.: *Communication and Concurrency*. Prentice Hall, (1989).
15. Milner, R.: *The Polyadic π -Calculus: a Tutorial*. Tech. Rep. ECS-LFCS-91-180, Laboratory for Foundations of Computer Science, Department of Computer Science, University of Edinburgh, UK (1991)
16. Park, D.: *Concurrency and Automata on Infinite Sequences*. In: LNCS 104. Springer-Verlag, (1981)
17. Roggenbach, M., Majster-Cederbaum, M.: Towards a Unified View of Bisimulation: a Comparative Study. *TCS*, 238, 81–130 (2000)
18. ITU-T. Z.120 Recommendation Z.120 (11/99): *Languages for telecommunications applications – Message Sequence Charts (MSC)* (1999)
19. ITU-T. Z.100 Recommendation Z.100 – *Specification and Description Language (SDL)* (1999)

20. Rutten, J.: Coalgebras and Systems. *TCS*, 249
21. Needham, R., Schroeder, M.: Using Encryption for Authentication in Large Networks of computers. *Comm. ACM*, 21(12), 993–999 (1978)
22. Lowe, G.: An Attack on the Needham-Schroeder Public Key Authentication Protocol. *Information Processing Letters*, 56(3), 131–136 (1995)
23. Eckerson, W.: Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications. *Open Information Systems*. 3(20), 10, 1 (1995)

Clocks Model for Specification and Analysis of Timing in Real-Time Embedded Systems

Iryna Zaretska¹, Galyna Zholtkevych¹, Grygoriy Zholtkevych¹
and Frédéric Mallet²

¹ V.N. Karazin Kharkiv National University,
School of Mathematics and Mechanics, 4, Svobody Sqr., 61022, Kharkiv, Ukraine

`zar@univer.kharkov.ua, {galyna,g}.zholtkevych@gmail.com`

² Université Nice Sophia Antipolis, AOSTE Team Project (INRIA/I3S),
INRIA Sophia Antipolis Méditerranée, 2004 rte des Lucioles (Lagrange L-043) BP93,
F-06902 Sophia Antipolis Cedex, France

`frederic.mallet@unice.fr`

Abstract. Problems concerning formal semantics for Clock Constraint Specification Language (CCSL) are considered in the paper. CCSL is intended for describing logical time models for real-time embedded systems and the language is a part of UML profile for MARTE. There exist two approaches to introduce a denotational semantics for CCSL. A pure relational subset of CCSL is defined in the paper. The notion of time structure with clocks is introduced to refine describing denotational semantics for this CCSL subset, which authors called RCCSL. Semantic properties of RCCSL have been studied. Theorem about coincidence semantics of RCCSL for the two approaches is proved.

Keywords. Embedded system, real-time system, time modelling, time structure, clock constraint, formal specification

Key terms. ConcurrentComputation, FormalMethod, SpecificationProcess, VerificationProcess, MathematicalModeling

1 Introduction

Nowadays, the growth of using distributed real-time systems (including embedded systems) [4] is the developing trend for Information and Communication Technology. There are two reasons for such growth: first, the physical limit for processor acceleration is reached, and, second, using mobile and cloud technologies are explosively expanded. The impossibility to continue over-clocking of a processor leads to using a multi-core system, which is parallel and distributed. A complex consisting of a computational cloud and an ensemble of mobile devices is a parallel and distributed system too. Moreover its structure is not fixed.

Each of the cases requires using different kinds of multiprocessing architectural and software solutions [3]. Therefore, providing correct working of such systems requires more research in the area.

Mathematical modelling of systems makes possible to develop formal specifications and methods of their analysis as a base for trustworthy system constructing. There are a lot of approaches to modelling multiprocessor systems. First of all, the following ones should be noticed: CSP of C.A.R. Hoare [8], π -calculus of R. Milner [11], abstract state machine model [5], and processing algebra [14].

This paper is devoted to formal methods for an important subclass of multiprocessing distributed systems, namely, real-time embedded (RTE) systems. These methods are closely connected with the UML profile for MARTE (Modelling and Analysis of Real-Time and Embedded systems) [2, 15]. In the context of the MARTE approach UML [16, 17] is used to build engineering models of a developing system. But the UML notation does not support detailed description of interactions for joining components into a united RTE system. A very common way to specify conditions for the system integrity is through the Object Constraint Language (OCL) [12]. However, no facilities for specifying temporal constraints are provided by the OCL standard. The Clock Constraint Specification Language (CCSL) [2] was defined in an annex of MARTE as a way to build logical and temporal constraints on model elements.

CCSL is intended to describe the temporal ordering of interactions between components of a distributed software system. It focuses on the ordering of event occurrences, but not on their chronometric characteristics. It relies on a logical time model inspired by the work on synchronous systems and their polychronous extensions.

The denotational semantics for basic constructions of CCSL is given in [10]. It is based on the notion of a time structure with clocks, other approach [1] defines an operational way to compute runs for CCSL specifications. The main contribution of this paper is a demonstration that the relationship of semantics consequence based on time structures as models of constraints and semantics consequence based on time structures associated with runs are only equivalent for a subset of CCSL, which we call RCCSL.

2 Syntax of Pure Relational CCSL

In the paper we restrict ourself to a very simple sublanguage of CCSL, which we call the pure relational CCSL (RCCSL). Syntax of this subset is given here using EBNF [6].

```

clock constraint =
  clock relation, {'', ' ', clock relation};
clock relation =
  clock reference, sign of clock relation, clock reference;
sign of clock relation =
  'subclocking' |

```

```

'exclusion'           |
'coincidence'        |
'cause'              |
'precedence';
clock reference =
  ? any element of clock set ?;

```

Below we use the next notation for symbols of clock relations (see Table 1).

Table 1. Symbols of clock relations

Relation name	Relation symbol
subclocking	\sqsubset
exclusion	$\#$
coincidence	\equiv
cause	\prec
precedence	\prec

These five binary relations on a clock set \mathcal{C} are determined as logical primitives for CCSL in [1].

Defining semantics for RCCSL is one of the paper objectives. Following the paper [10], we define the denotational meaning for a set of clock constraints as some class of time structures expanded by a classification for event occurrences. The next section is devoted to describing such structures.

3 Time Structure with Clocks

Let consider a set of event occurrences, which is below denoted by \mathcal{I} . Elements of the set \mathcal{I} are called instants. Some pairs of instants denotes instant pairs, whose elements are ordered in time: $i_1 \prec i_2$ is denoted the fact "an instant i_1 causes an instant i_2 " or equivalently "an instant i_1 cannot occur later than an instant i_2 ", where $i_1, i_2 \in \mathcal{I}$. This relation is called 'cause'. It is naturally to suppose that cause is a pre-order.

As known [7, section 1.3], each pre-order can be decomposed uniquely into the union of two relations such that the former is a strict order (it is denoted below by ' $<$ ' and called a precedence) and the latter is an equivalence (it is denoted below by ' \equiv ' and called a coincidence). These relations are connected by the next property:

$$\begin{aligned}
 &\text{for any instants } i_1, i'_1, i_2, i'_2 \in \mathcal{I} \\
 &\text{the validity of } i_1 \equiv i'_1, i_2 \equiv i'_2, \text{ and } i_1 < i_2 \text{ implies} \\
 &\text{truth of } i'_1 < i'_2.
 \end{aligned} \tag{1}$$

Moreover, if we have a strict order and an equivalence on the same set and these relations satisfy (1) then their union is a pre-order.

Now, we can introduce the notion of a time structure for formalising our understanding a set of instants.

Definition 1. *Let (\mathcal{I}, \preceq) be a pair of a set and a pre-order on this set respectively. Denote by \prec the strict order corresponding to the pre-order \preceq . The pair (\mathcal{I}, \preceq) is called a time structure if the next property (the property of cause finiteness [13]) holds:*

$$\text{the set } \{i' \in \mathcal{I} \mid i' \prec i\} \text{ is finite for all } i \in \mathcal{I}. \tag{2}$$

Definition 1 is based on the corresponding definition in [10]. One can compare them with the definition of a time structure in [13]. Difference consists in a possibility of modelling an instant coincidence.

Note that Definition 1 specifies the set of instants and some time relations on it but it does not determine any classification of instants in compliance with their sources. Therefore, in the following [2] we introduce such a classification by adding a finite set of instant sources called clocks and by mapping the set of instants into this clock set.

Definition 2. *Let (\mathcal{I}, \preceq) be a time structure, \mathcal{C} be a finite set of clocks, and $\pi : \mathcal{I} \rightarrow \mathcal{C}$ be a map then the quadruple $(\mathcal{I}, \preceq, \mathcal{C}, \pi)$ is called a time structure with clocks if the next property holds:*

$$\begin{aligned} &\text{for any clock } c \in \mathcal{C} \text{ and } i_1, i_2 \in \pi^{-1}(c) \\ &\text{the validity of } i_1 \neq i_2 \text{ implies truth of } i_1 \prec i_2 \vee i_2 \prec i_1, \\ &\text{i.e. } \pi^{-1}(c) \text{ is linearly ordered by the restriction of the cause.} \end{aligned} \tag{3}$$

If $c \in \mathcal{C}$ then the set $\pi^{-1}(c)$ is usually denoted by \mathcal{I}_c . It can be considered as an event stream generated by the source associated with the clock c .

From Definition 1 and Definition 2 the next fact follows immediately.

Proposition 1. *Let $(\mathcal{I}, \preceq, \mathcal{C}, \pi)$ be a time structure with clocks then*

1. \mathcal{I}_c is well-ordered by the strict order \prec for all $c \in \mathcal{C}$;
2. ordinal type of \mathcal{I}_c for any $c \in \mathcal{C}$ is less or equal to ω , where ω is the first infinite ordinal.

Proof. Firstly note that property (3) implies linear ordering \mathcal{I}_c for an arbitrary $c \in \mathcal{C}$.

Further, suppose that A is some non-empty subset of \mathcal{I}_c for an arbitrary $c \in \mathcal{C}$, i is some element of A .

If for all $i' \in A$ the statement $i \prec i' \vee i = i'$ is true then $\inf A = i \in A$.

If there exists $i_0 \in A$ such that $i_0 \prec i$ then the set $A(i) = \{i' \in A \mid i' \prec i\}$ is not empty. It is evident that $A(i) = A \cap \{i' \in \mathcal{I}_c \mid i' \prec i\}$. This equality and the property of cause finiteness (2) imply finiteness of $A(i)$. So, taking into account

property (3) we can conclude that $A(i)$ is a finite linearly ordered set. Hence, there exists $i_* \in A(i)$ such that $i_* = \inf A(i)$. It is evident that

$$\inf A = \inf A(i) = i_* \in A(i) \subset A.$$

Thus, $\inf A \in A$ and \mathcal{I}_c is well-ordered.

The supposition that ordinal type of \mathcal{I}_c for some $c \in \mathcal{C}$ is greater than ω is inconsistent with the property of cause finiteness (2). \square

Corollary 1. *Any instant $i \in \mathcal{I}$ is uniquely determined by the pair $(\pi(i), \text{idx}(i))$, which is an element of the set $\mathcal{C} \times \mathbb{N}$. Here, idx is a map from \mathcal{I} into \mathbb{N} such that*

$$\text{idx}(i) = |\{i' \in \mathcal{I}_{\pi(i)} \mid i' \prec i\}| + 1,$$

where the number of elements in a set A is denoted by $|A|$.

The designation $\mathfrak{T}^{\mathcal{C}}$ is used below to refer to the class of time structures with \mathcal{C} as a set of clocks.

Remark 1. One can show that this class is a set but we do not do it in the paper.

4 Denotational Semantics for RCCSL

Usually, a denotational semantics can be considered as the theory of models for the corresponding language. We shall use time structures with clocks as models for describing meaning of clock constraints.

4.1 Some General Notes

One can identify a class of event occurrences of the same type with a set of instants for some clock in the process of specifying interactions between components of distributed parallel systems. Such an identification is provided by fixing a set of clocks \mathcal{C} and describing rules of interacting system components. These rules divide the set $\mathfrak{T}^{\mathcal{C}}$ into two subsets: the subset of time structures satisfying the constraints and the set of time structures contradicting them. Taking into account the specification of RCCSL one can say that a clock constraint is a finite set of clock relations. If the set of clock relations determining the constraint is denoted by \mathfrak{C} then the fact "the time structure $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ satisfies the constraint \mathfrak{C} " can be written as $\mathcal{T} \models \mathfrak{C}$. More precisely, $\mathcal{T} \models \mathfrak{C}$ means that for each $C \in \mathfrak{C}$ the clause $\mathcal{T} \models C$ is true.

Further, for a constraint \mathfrak{C} , $\llbracket \mathfrak{C} \rrbracket$ denote the following set $\{\mathcal{T} \in \mathfrak{T}^{\mathcal{C}} \mid \mathcal{T} \models \mathfrak{C}\}$.

The first important problem is the consistency problem for the constraint. The rigorous problem formulation has usually the form:

Problem 1 (Consistency Problem). For a constraint \mathfrak{C} check that the set $\llbracket \mathfrak{C} \rrbracket$ is not empty.

The second important problem is the semantic consequence for the constraints. The rigorous problem formulation has the next form:

Problem 2 (Semantic Consequence Problem). For a constraint \mathfrak{C} and a clock relation τ check that $\llbracket \mathfrak{C} \rrbracket \subset \llbracket \tau \rrbracket$ (or in the another notation $\mathfrak{C} \Vdash \tau$).

Below we use the notation $\{\mathfrak{C}\}$ for the set of clock relations that form the constraint \mathfrak{C} . It is easy to see that the next properties of the relationship \Vdash are true.

Proposition 2. *The next properties are satisfied:*

1. if a constraint \mathfrak{C} and a clock relation τ satisfy the condition $\tau \in \{\mathfrak{C}\}$ then $\mathfrak{C} \Vdash \tau$;
2. if constraints \mathfrak{C}_1 and \mathfrak{C}_2 and a clock constraint τ satisfy the next condition $\mathfrak{C}_1 \Vdash \tau'$ for all $\tau' \in \{\mathfrak{C}_2\}$ and $\mathfrak{C}_2 \Vdash \tau$ are true then $\mathfrak{C}_1 \Vdash \tau$ is true.

Proof is omitted □

To complete defining the denotational semantics for RCCSL we should determine the meaning of basic clock relations.

4.2 Subclocking

This relation is intended for specifying a requirement to synchronize each instant of one clock with some instant of another clock. In this case the first clock is called a subclock of the second clock.

More precisely, let $c', c'' \in \mathcal{C}$ and $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ then $\mathcal{T} \models c' \sqsubset c''$ means that there exists a strict monotonic map $h : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$ such that $i \equiv h(i)$ for any $i \in \mathcal{I}_{c'}$.

Proposition 3 (Trivial Subclocking). *For each $c \in \mathcal{C}$ the clause $\Vdash c \sqsubset c$ is true.*

Proof is trivial □

Proposition 4 (Transitivity Law for Subclocking). *For each $c', c'', c''' \in \mathcal{C}$ the clause $c' \sqsubset c'', c'' \sqsubset c''' \Vdash c' \sqsubset c'''$ is true.*

Proof. Let $h_{c'' c'} : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$, $h_{c''' c''} : \mathcal{I}_{c''} \rightarrow \mathcal{I}_{c'''}$ be strict monotonic maps providing the validity of the clauses $\mathcal{T} \models c' \sqsubset c''$ and $\mathcal{T} \models c'' \sqsubset c'''$ respectively for some \mathcal{T} . It is easy to see that the map $h_{c''' c''} \circ h_{c'' c'}$ provides the validity of the clause $\mathcal{T} \models c' \sqsubset c'''$ □

4.3 Exclusion

This relation is used for specifying the mutual exclusion for two events.

More formally, let $c', c'' \in \mathcal{C}$ and $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ then $\mathcal{T} \models c' \# c''$ means that for any $i' \in \mathcal{I}_{c'}$, $i'' \in \mathcal{I}_{c''}$ the coincidence $i' \equiv i''$ is false.

Proposition 5 (Irreflexivity Law for Exclusion). *For each $c \in \mathcal{C}$ the equality $\llbracket c \boxminus c \rrbracket = \emptyset$ is true.*

Proof is trivial □

Proposition 6 (Symmetry Law for Exclusion). *For each $c', c'' \in \mathcal{C}$ the clause $c' \boxminus c'' \Vdash c'' \boxminus c'$ is true.*

Proof is trivial □

4.4 Coincidence

This relation describes synchronization of two event sources.

More precisely, let $c', c'' \in \mathcal{C}$ and $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ then $\mathcal{T} \models c' \boxplus c''$ means that there exists a strict monotonic bijection $h : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$ such that $i \equiv h(i)$ for any $i \in \mathcal{I}_{c'}$.

Proposition 7 (Trivial Coincidence). *For each $c \in \mathcal{C}$ the clause $\Vdash c \boxplus c$ is true.*

Proof is trivial □

Proposition 8 (Symmetry Law for Coincidence). *For each $c', c'' \in \mathcal{C}$ the clause $c' \boxplus c'' \Vdash c'' \boxplus c'$ is true.*

Proof. Let $h : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$ be a strict monotonic bijection providing the validity of the clause $\mathcal{T} \models c' \boxplus c''$ for some \mathcal{T} and h^{-1} be its inverse map. Suppose that $i', i'' \in \mathcal{I}_{c''}$, $i' \prec i''$, and $h^{-1}(i') \not\prec h^{-1}(i'')$ then either $h^{-1}(i') = h^{-1}(i'')$ or $h^{-1}(i'') \prec h^{-1}(i')$. But the first alternative contradicts to bijectivity of h , and the second alternative and strict monotonicity of h implies $i'' \prec i'$. The last clause contradicts to irreflexivity of the precedence relation. These contradictions show that h^{-1} is a strict monotonic map.

Further, for any $i \in \mathcal{I}_{c''}$ we have that $h^{-1}(i) \in \mathcal{I}_{c'}$ and $h^{-1}(i) \equiv h(h^{-1}(i)) = i$. Thus, the clause $\mathcal{T} \models c'' \boxplus c'$ is true □

Proposition 9 (Transitivity Law for Coincidence). *For each $c', c'', c''' \in \mathcal{C}$ the clause $c' \boxplus c'', c'' \boxplus c''' \Vdash c' \boxplus c'''$ is true.*

Proof is similar to proof of Proposition 4 □

4.5 Cause

This relation is intended for specifying that each instant of one clock is caused by an instant in another clock.

More precisely, let $c', c'' \in \mathcal{C}$ and $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ then $\mathcal{T} \models c' \boxdot c''$ means that there exists a strict monotonic map $h : \mathcal{I}_{c''} \rightarrow \mathcal{I}_{c'}$ such that $h(i) \preceq i$ for any $i \in \mathcal{I}_{c''}$.

Proposition 10 (Trivial Cause). *For each $c \in \mathcal{C}$ the clause $\Vdash c \boxdot c$ is true.*

Proof is trivial □

Proposition 11 (Transitivity Law for Cause). *For each $c', c'', c''' \in \mathcal{C}$ the clause $c' \boxed{\preceq} c'', c'' \boxed{\preceq} c''' \Vdash c' \boxed{\preceq} c'''$ is true.*

Proof is similar to proof of Proposition 4 □

4.6 Precedence

This relation is a stronger variant of the cause relation.

Namely, let $c', c'' \in \mathcal{C}$ and $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ then $\mathcal{T} \models c' \boxed{\prec} c''$ means that there exists a strict monotonic map $h : \mathcal{I}_{c''} \rightarrow \mathcal{I}_{c'}$ such that $h(i) \prec i$ for any $i \in \mathcal{I}_{c''}$.

Proposition 12 (Irreflexivity Law for Precedence). *For each $c \in \mathcal{C}$ the equality $\llbracket c \boxed{\prec} c \rrbracket = \emptyset$ is true.*

Proof is trivial □

Proposition 13 (Transitivity Law for Precedence). *For each $c', c'', c''' \in \mathcal{C}$ the clause $c' \boxed{\prec} c'', c'' \boxed{\prec} c''' \Vdash c' \boxed{\prec} c'''$ is true.*

Proof is similar to proof of Proposition 4 □

4.7 Interdependencies Laws for the Basic Relations

Above we considered properties of each basic relation but interdependencies between these relations were not in our focus. Thus, such interdependencies are considered below. The next lemma is needed to ground these dependencies.

Lemma 1. *Let (X, \leq) be a well-ordered set and $\phi : X \rightarrow X$ be a strict monotonic map such that for all $x \in X$ the assertion $\phi(x) \leq x$ is true then ϕ is the identity map.*

Proof. One can prove the lemma by using the transfinite induction □

Proposition 14 (Interdependencies Laws for the Basic Relations).

1. For each $c', c'' \in \mathcal{C}$ the clause $c' \boxed{\sqsubset} c'', c'' \boxed{\sqsubset} c' \Vdash c' \boxed{\equiv} c''$ is true.
2. For each $c', c'' \in \mathcal{C}$ the clock relations $c' \boxed{\sqsubset} c''$ and $c' \boxed{\#} c''$ are inconsistent, i.e. $\llbracket c' \boxed{\sqsubset} c'', c' \boxed{\#} c'' \rrbracket = \emptyset$.
3. For each $c', c'' \in \mathcal{C}$ the clause $c' \boxed{\sqsubset} c'' \Vdash c'' \boxed{\preceq} c'$ is true.
4. For each $c', c'' \in \mathcal{C}$ the clause $c' \boxed{\preceq} c'', c'' \boxed{\preceq} c' \Vdash c' \boxed{\equiv} c''$ is true.

Proof. 1) For any $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ the validity of the assertion " $\mathcal{T} \models c' \boxed{\equiv} c''$ " implies $\mathcal{T} \models c' \boxed{\sqsubset} c''$ is evident.

Let's check the validity of the inverse assertion. Denote the strict monotonic maps that provide for some $\mathcal{T} \in \mathfrak{T}^{\mathcal{C}}$ the validity of $\mathcal{T} \models c' \boxed{\sqsubset} c''$ and $\mathcal{T} \models c'' \boxed{\sqsubset} c'$

by $h_{c' c'} : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$ and $h_{c' c''} : \mathcal{I}_{c''} \rightarrow \mathcal{I}_{c'}$ respectively. We claim that they are mutually inverse.

Indeed, for any $i \in \mathcal{I}_{c'}$ we have the next coincidences: $i \equiv h_{c' c''}(i)$ and $h_{c' c''}(i) \equiv h_{c' c''}(h_{c' c''}(i))$. These coincidences and the Transitivity Law for Coincidence (see Proposition 9) provide the validity of the coincidence $i \equiv h_{c' c''}(h_{c' c''}(i))$. Taking into account that both i and $h_{c' c''}(h_{c' c''}(i))$ are elements of $\mathcal{I}_{c'}$ and the fact that restriction of \preceq on $\mathcal{I}_{c'}$ is a strict order (see Proposition 1) one can derive the equality $i = h_{c' c''}(h_{c' c''}(i))$.

The equality $i = h_{c' c''}(h_{c' c''}(i))$ for all $i \in \mathcal{I}_{c''}$ is derived similarly. Thus, $h_{c' c'}$ is a bijection.

2) Proof is trivial.

3) Proof is trivial.

4) Really, let $h_{c',c'} : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c''}$ and $h_{c',c''} : \mathcal{I}_{c''} \rightarrow \mathcal{I}_{c'}$ be strict monotonic maps provided for some $\mathcal{T} \in \mathfrak{T}^{\mathcal{T}}$ the validity of the clauses $\mathcal{T} \models c'' \sqsubseteq c'$ and $\mathcal{T} \models c' \sqsubseteq c''$ respectively. Then the map $\phi = h_{c',c''} \circ h_{c',c'} : \mathcal{I}_{c'} \rightarrow \mathcal{I}_{c'}$ is strict monotonic and it satisfies the condition $\phi(i) \preceq i$. Therefore, applying the Lemma 1 allows to conclude that ϕ and the identity map are equal \square

5 Runs and Chronometers

Following [1], in this section we introduce the notion of a run for a set of clocks. We use this notion to define a behavioural model for the set of clocks.

Definition 3 (see [1]). *Let \mathcal{C} be a finite set of clock then any map $\mathbf{r} : \mathbb{N} \rightarrow 2^{\mathcal{C}}$ such that $\mathbf{r}(t) = \emptyset$ implies $\mathbf{r}(t') = \emptyset$ for all $t' > t$ is called a run for \mathcal{C} .*

This definition means that if \mathbf{r} is a run then at the (global) time t all clocks of the set $\mathbf{r}(t)$ and only them are triggered.

For each run \mathbf{r} one can construct a quadruple $\mathcal{T}[\mathbf{r}] = (\mathcal{I}_{\mathbf{r}}, \preceq, \mathcal{C}, \pi_{\mathbf{r}})$ by the following way:

- $\mathcal{I}_{\mathbf{r}} = \{(c, t) \in \mathcal{C} \times \mathbb{N} \mid c \in \mathbf{r}(t)\}$;
- $(c', t') \preceq (c'', t'')$ if and only if $t' \leq t''$;
- $\pi_{\mathbf{r}}(c, t) = c$ for all $(c, t) \in \mathcal{I}_{\mathbf{r}}$.

Proposition 15. $\mathcal{T}[\mathbf{r}]$ is a time structure with clocks for given run \mathbf{r} .

Proof. It is proved by trivial checking properties (2) and (3) \square

Hence, we can define the semantic relationship between a run \mathbf{r} and a constraint \mathfrak{C} by the next way: $\mathbf{r} \models \mathfrak{C}$ if and only if the clause $\mathcal{T}[\mathbf{r}] \models \mathfrak{C}$ is true. Also, we can introduce the relationship $\mathfrak{C}_1 \Vdash_{run} \mathfrak{C}_2$ as an abbreviation of the sentence "for any \mathbf{r} such that $\mathbf{r} \models \mathfrak{C}_1$ the next relationship $\mathbf{r} \models \mathfrak{C}_2$ is valid".

Proposition 15 allows to suggest that a run carries more information than a time structure because a run depends on global time. A refinement and a substantiation of this hypothesis is discussed below.

The notion of chronometer is introduced to specify dependences between time structures and runs.

Definition 4. Let $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$ be a time structure with clocks and $\chi : \mathcal{I} \rightarrow \mathbb{N}$ be a map such that the next assertions are true:

$$\text{for any } i', i'' \in \mathcal{I} \text{ the coincidence } i' \equiv i'' \text{ implies } \chi(i') = \chi(i'') \quad (4)$$

$$\text{for any } i', i'' \in \mathcal{I} \text{ the strict precedence } i' \prec i'' \text{ implies } \chi(i') < \chi(i'') \quad (5)$$

$$\text{for any } t, t' \in \mathbb{N} \text{ the validity of the clauses } t \in \chi(\mathcal{I}) \text{ and } t' < t \text{ implies truth of the clause } t' \in \chi(\mathcal{I}) \quad (6)$$

then χ is called a chronometer on \mathcal{T} [9].

Example 1. Let \mathcal{C} be a finite set of clocks, \mathbf{r} be a run for \mathcal{C} . Then it is evident that the map $\chi_* : \mathcal{I}_{\mathbf{r}} \rightarrow \mathbb{N}$ determined by the equality $\chi_*(c, t) = t$ is a chronometer.

Hence, Example 1 shows that each time structure generated by a run has a native chronometer χ_* .

Proposition 16. Let \mathcal{T} be a time structure with clocks and $\chi : \mathcal{I} \rightarrow \mathbb{N}$ be a chronometer then the map $\mathbf{r}[\mathcal{T}, \chi] : \mathbb{N} \rightarrow \mathbf{2}^{\mathcal{C}}$ defined by the next formula

$$\mathbf{r}[\mathcal{T}, \chi](t) = \pi(\chi^{-1}(t)) \quad (7)$$

is a run.

Proof. To prove the proposition we should show that $\mathbf{r}[\mathcal{T}, \chi](t) = \emptyset$ for some $t \in \mathbb{N}$ implies $\mathbf{r}[\mathcal{T}, \chi](t') = \emptyset$ for any $t' \geq t$.

Suppose existence of t_1 and t_2 such that $t_1 < t_2$, $\pi(\chi^{-1}(t_1)) = \emptyset$, but $\pi(\chi^{-1}(t_2)) \neq \emptyset$. Taking into account this assumption one can derive that $\chi^{-1}(t_1) = \emptyset$ and $\chi^{-1}(t_2) \neq \emptyset$. Hence, $t_1 \notin \chi(\mathcal{I})$ and $t_2 \in \chi(\mathcal{I})$. We have obtained the contradiction to condition (6) of Definition 4 \square

The next property for the chronometer χ_* from Example 1 holds.

Proposition 17. Let \mathbf{r} be a run for a clock set \mathcal{C} then the next equality holds

$$\mathbf{r}[\mathcal{T}[\mathbf{r}], \chi_*] = \mathbf{r}. \quad (8)$$

Let $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$ be a time structure with clocks and $\chi : \mathcal{I} \rightarrow \mathbb{N}$ be a chronometer on \mathcal{T} then the map $\widehat{\chi} : \mathcal{I} \rightarrow \mathcal{C} \times \mathbb{N}$ defined in the next way $\widehat{\chi}(i) = (\pi(i), \chi(i))$ is a map onto $\mathcal{I}_{\mathbf{r}[\mathcal{T}, \chi]}$ such that any coincidence $i' \equiv i''$ implies the coincidence $\widehat{\chi}(i') \equiv \widehat{\chi}(i'')$ in $\mathcal{T}[\mathbf{r}]$ and any precedence $i' \prec i''$ implies the precedence $\widehat{\chi}(i') \prec \widehat{\chi}(i'')$ in $\mathcal{T}[\mathbf{r}]$.

Proof. Really,

$$\begin{aligned} \mathbf{r}[\mathcal{T}[\mathbf{r}], \chi_*](t) &= \pi_{\mathbf{r}}(\chi_*^{-1}(t)) = \\ &= \pi_{\mathbf{r}}(\{(c, t) \in \mathcal{I}_{\mathbf{r}}\}) = \pi_{\mathbf{r}}(\{(c, t) \in \mathcal{C} \times \mathbb{N} \mid c \in \mathbf{r}(t)\}) = \mathbf{r}(t). \end{aligned}$$

Further, $(c, t) \in \mathcal{I}_{\mathbf{r}[\mathcal{T}, \chi]}$ if and only if $c \in \mathbf{r}[\mathcal{T}, \chi](t)$. It is easy to see that the last clause is equivalent to existence of $i \in \mathcal{I}$ such that $c = \pi(i)$ and $t = \chi(i)$, i.e. it is equivalent to $(c, t) = \widehat{\chi}(i)$.

If $i' \equiv i''$ then $\chi(i') = \chi(i'')$ by definition of a chronometer, hence $\widehat{\chi}(i') \equiv \widehat{\chi}(i'')$. Similarly, if $i' \prec i''$ then $\chi(i') < \chi(i'')$, therefore $\widehat{\chi}(i') \prec \widehat{\chi}(i'')$ \square

Proposition 18. *There exists only one chronometer on $\mathcal{T}[\mathbf{r}]$ for any run \mathbf{r} .*

Proof. For any run \mathbf{r} there exists the chronometer χ_* on $\mathcal{T}[\mathbf{r}]$. Let χ be an other chronometer on $\mathcal{T}[\mathbf{r}]$. For $(c', t), (c'', t) \in \mathcal{I}[\mathbf{r}]$ using (4) we have $\chi(c', t) = \chi(c'', t)$. Hence, taking into account Definition 3 one can obtain that $\chi(c, t) = \tau(t)$ where τ is strict monotonic function from α into α for some cardinal $\alpha \leq \omega$. Thus, τ is the identity function and $\chi = \chi_*$ \square

Hence, a chronometer exists on a time structure associated with a run. We claim that a chronometer exists on any time structure with clocks.

The next binary relation \triangleleft on a time structure with clocks will be used for describing an algorithm that calculates timestamps for instants. More precisely, if $i', i'' \in \mathcal{I}$ then $i' \triangleleft i''$ means that for all $i \in \mathcal{I}$ the validity of the next clause $i \prec i'' \ \& \ i' \preceq i$ implies truth of the coincidence $i \equiv i'$. It is easy seen that if $i_1 \equiv i'_1, i_2 \equiv i'_2$, and $i_1 \triangleleft i_2$ then $i'_1 \triangleleft i'_2$.

Now we can construct the algorithm that allows to calculate timestamps for instants on an arbitrary time structure with clocks. This Algorithm 1 is a generalization of Lamport's algorithm [9].

Algorithm 1: Computing timestamp for an instant

```

input :  $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$  is a time structure with clocks,
          $i$  is an element of  $\mathcal{I}$ 
output: timestamp for the instant  $i$ 

1 begin
2   count  $\leftarrow$  1;  $D \leftarrow \emptyset$ ;  $W \leftarrow \emptyset$ ;      // -- initializing work variables --
3   while  $i \notin D$  do                                     // -- main loop -----
4      $W_+ \leftarrow \{j \in \mathcal{I} \mid j \notin D \ \& \ \text{idx}(j) = \text{count}\}$ ;
5      $W_+ \leftarrow W_+ \cup \{j \in \mathcal{I} \mid j \notin D \ \& \ (\exists j' \in W_+) j' \equiv j\}$ ;
6      $W \leftarrow W \cup W_+$ ;
7      $D_+ \leftarrow \{j \in W \mid (\forall j' \in \mathcal{I})(j' \triangleleft j \Rightarrow j' \in D)\}$ ;
8      $D \leftarrow D \cup D_+$ ;
9      $W \leftarrow W \setminus D_+$ ;
10    count  $\leftarrow$  count + 1;
11  end
12  return count;
13 end

```

Theorem 1 (existence of a chronometer). *Let \mathcal{T} be a time structure with clocks and $\chi_0 : \mathcal{I} \rightarrow \mathbb{N}$ be the function calculated by Algorithm 1 then χ_0 is a chronometer on \mathcal{T} .*

Proof. One can see that Algorithm 1 builds two sequences of sets

$$D_0 \subset D_1 \subset D_2 \subset \dots \subset D_n \subset \dots$$

$$W_0, W_1, W_2, \dots, W_n, \dots$$

in accordance to the following computational scheme:

$$\begin{cases} W_0 &= \emptyset \\ D_0 &= \emptyset \\ W_{n+1} &= (W_n \cup \{j \in \mathcal{I} \mid (\exists j' \in \mathcal{I})(j' \equiv j \ \& \ \text{idx}(j') = n + 1)\}) \setminus D_n \\ D_{n+1} &= D_n \cup \{j \in W_{n+1} \mid (\forall j' \in \mathcal{I})(j' \triangleleft j \Rightarrow j' \in D_n)\} \end{cases}$$

and maps an instant $i \in \mathcal{I}$ into $\chi_0(i) = \inf\{n \in \mathbb{N} \mid i \in D_n\}$.

Firstly, note that supposition about partial definiteness of χ_0 implies existence of an infinite sequence $i_1 \triangleright i_2 \triangleright \dots$. But it contradicts the causes finiteness property (2).

Secondly, it is true by the construction of D_n that the validity of $i' \equiv i''$ implies the truth of the following statement: $i' \in D_n$ if and only if $i'' \in D_n$. Hence, we obtain that $i' \equiv i''$ implies $\chi_0(i') = \chi_0(i'')$.

Further, similar reasoning provides the validity of the following statement: $i' \prec i''$ implies $\chi_0(i') < \chi_0(i'')$.

Finally, the simple inequality $\text{idx}(i) \leq \chi(i)$, which is correct for any $i \in \mathcal{I}$ and any chronometer χ on \mathcal{T} , provides the validity of property (6) \square

Corollary 2. *There exists a chronometer on an arbitrary time structure with clocks.*

6 Equivalence of Semantics for RCCSL Determined by Relations \Vdash and \Vdash_{run}

In the section the notion of a chronometer is used to prove the theorem about equivalence of the relationships \Vdash and \Vdash_{run} . The theorem is the main result of the paper. Taking into account the theorem one can confine himself to checking semantic consequence by using runs. This opens a way to constructing an operational semantics of RCCSL so that it is equivalent to the denotational semantics defined above.

We need two lemmas to prove the main theorem.

Let's use the notation $i_1 \parallel i_2$ for instants i_1 and i_2 such that $i_1 \not\prec i_2$ & $i_2 \not\prec i_1$.

Lemma 2. *Let $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$ be a time structure with clocks and i_1, i_2 be instants such that the clause $i_1 \parallel i_2$ is true then there exists a chronometer χ on \mathcal{T} satisfied the following condition $\chi(i_1) < \chi(i_2)$.*

Proof. Let's consider the quadruple $\mathcal{T}' = (\mathcal{C}, \mathcal{I}, \preceq', \pi)$ such that $i' \prec' i''$ is valid if one of the next conditions is true

1. $i' = i_1$ and $i'' = i_2$;
2. $i' \prec i''$;
3. $i' \prec i_1$ and $i_2 \prec i''$;

and $i' \preceq' i''$ if and only if $i' \equiv i''$ or $i' \prec' i''$. It is easy seen that the relation \preceq' is a pre-order. More over, it satisfies properties (2) and (3). Hence, \mathcal{T}' is

a time structure with clocks. Using Corollary 2 we obtain that there exists a chronometer χ on \mathcal{T}' . But then χ is a chronometer on \mathcal{T} and $\chi(i_1) < \chi(i_2)$ is true \square

Corollary 3. *Let $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$ be a time structure with clocks, $i', i'' \in \mathcal{I}$ be instants, then*

1. $i' \prec i''$ is valid if and only if for any chronometer χ on \mathcal{T} the inequality $\chi(i') < \chi(i'')$ is true;
2. $i' \equiv i''$ is valid if and only if for any chronometer χ on \mathcal{T} the equality $\chi(i') = \chi(i'')$ is true.

Lemma 3. *Let $\mathcal{T} = (\mathcal{C}, \mathcal{I}, \preceq, \pi)$ be a time structure with clocks, $\boxed{*}$ be an arbitrary sign of a clock relation, c' and c'' be clocks then $\mathcal{T} \models c' \boxed{*} c''$ if and only if $\mathbf{r}[\mathcal{T}, \chi] \models c' \boxed{*} c''$ for any chronometer χ on \mathcal{T} .*

Proof. It is evident that $\mathcal{T} \models c' \boxed{*} c''$ implies $\mathbf{r}[\mathcal{T}, \chi] \models c' \boxed{*} c''$ for any chronometer χ on \mathcal{T} . Hence, we need to prove the inverse statement.

1) Suppose that $\mathbf{r}[\mathcal{T}, \chi] \models c' \boxed{\subset} c''$ for any chronometer χ on \mathcal{T} . Then for any $i \in \mathcal{I}_{c'}$ and for each chronometer χ there exists an instant $i^\chi \in \mathcal{I}_{c''}$ such that $\chi(i) = \chi(i^\chi)$. Denote by X the set formed all i^χ . It is a nonempty subset of $\mathcal{I}_{c''}$. Suppose that there exists at least two different elements in the set X . Let's denote them by i^{χ_1} and i^{χ_2} . Taking in account linearity of the order on $\mathcal{I}_{c'}$ and $i^{\chi_1} \neq i^{\chi_2}$ one can suppose that $i^{\chi_1} \prec i^{\chi_2}$. Therefore $\chi_1(i) = \chi_1(i^{\chi_1}) < \chi_1(i^{\chi_2})$. Thus, one of the two cases is realised: $i \prec i^{\chi_2}$ or $i \parallel i^{\chi_2}$. But in the first case we obtain the inequality $\chi_2(i) < \chi_2(i^{\chi_2})$, which contradicts to the choice of i^{χ_2} . Hence, $i \parallel i^{\chi_2}$ is true. Similarly, one can obtain that $i \parallel i^{\chi_1}$ is true. Therefore, we proved that $|X| > 1$ implies $i \parallel i^\chi$ for all $i \in \mathcal{I}_{c'}$ and any chronometer χ .

Let $i^* = \inf_{\chi \in X} i^\chi$ then $i \parallel i^*$ and $\chi(i^*) \leq \chi(i^\chi) = \chi(i)$. This is a contradiction because Lemma 2 provides existence of some chronometer χ_0 such that $\chi_0(i^*) > \chi_0(i)$. Hence, X contains only one element, which we denote by $h(i)$. By construction we have $\chi(i) = \chi(h(i))$ for any chronometer χ . The last property implies strict monotonicity of h and the coincidence $i \equiv h(i)$. Therefore, $\mathcal{T} \models c' \boxed{\subset} c''$.

2 and 3) Suppose that $\mathbf{r}[\mathcal{T}, \chi] \models c' \boxed{*} c''$ for any chronometer χ on \mathcal{T} then it is evident that $\mathcal{T} \models c' \boxed{*} c''$ where $\boxed{*}$ equals to $\boxed{\#}$ or $\boxed{=}$.

4 and 5) Suppose that $\mathbf{r}[\mathcal{T}, \chi] \models c' \boxed{*} c''$ for any chronometer χ on \mathcal{T} where $\boxed{*}$ equals to $\boxed{\preceq}$ or $\boxed{\prec}$. Similarly, in the first case one can derive that $\mathcal{T} \models c' \boxed{*} c''$ is true \square

Theorem 2 (about equivalence of semantics). *Let \mathcal{C} be an arbitrary finite set of clocks, \mathfrak{C}_1 and \mathfrak{C}_2 be RCCSL constraints then the $\mathfrak{C}_1 \Vdash \mathfrak{C}_2$ is true if and only if $\mathfrak{C}_1 \Vdash_{run} \mathfrak{C}_2$ is true.*

Proof. One can easily see that the Theorem is the direct consequence of the Lemma 3 \square

7 Conclusion

In the paper we have considered the pure relational subset of CCSL (RCCSL) and have introduced semantics for it by using a class of mathematical objects called by authors time structures with clocks.

We have studied semantic properties of RCCSL (see Propositions 3 – 9). We hope that these properties can be a background of an axiomatic basis for analysing relational clock constraints.

Further we have introduced the notions "a run" and "a chronometer". It allowed us to study interrelations between time structures and runs, to introduce the alternative semantics closer to the operational approach than the denotational semantics discussed earlier.

Finally, the main theorem about equivalence of these two semantics (see Theorem 2) has been proved.

We are planning to continue our research in the next areas:

- building an axiomatic theory of the semantic consequence for RCCSL constraints;
- extending results on complete CCSL;
- studying an operational semantics of CCSL and specifying its interrelations to the denotational semantics.

References

1. André, C.: Syntax and Semantics of the Clock Constraint Specification Language (CCSL). Technical report, RR-6925, INRIA (2009), <http://hal.inria.fr/inria-00384077/en/>
2. André, C., Mallet, F., de Simone, R.: The Time Model of Logical Clocks available in the OMG MARTE profile. In: Shukla, S.K., Talpin, J.-P. (eds.) "Synthesis of Embedded Software: Frameworks and Methodologies Correctness by Construction", pp. 201–227. Springer Science+Business Media, LLC New York (2010)
3. Baer, J.-L.: Multiprocessing Systems. IEEE Trans. on Computers. 12, vol. C-25, 1271–1277 (1976)
4. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the first edition of the MCC workshop on Mobile cloud computing, pp. 13–16. ACM New York, NY, USA (2012)
5. Börger, E., Stärk, R.: Abstract State Machines: A Method for High-Level System Design and Analysis. Springer-Verlag, Berlin Heidelberg (2003)
6. Information technology – Syntactic metalanguage – Extended BNF. ISO/IEC 14977:1996(E)
7. Harzheim, E.: Ordered Sets. Springer Science+Business Media, Inc. New York (2005)
8. Hoare, C.A.R.: Communicating Sequential Processes. Prentice Hall International (1985)
9. Lamport, L.: Time, Clocks, and the Ordering of Events in a Distributed System. Comm. ACM. 7, vol. 12, 558–565 (1978)
10. Mallet, F.: Logical Time @ Work for the Modeling and Analysis of Embedded Systems, Habilitation thesis. LAMBERT Academic Publishing (2011)

11. Milner, R.: *Communicating and Mobile Systems: The Pi Calculus*. Cambridge University Press, Cambridge (1999)
12. Information technology – Object Management Group – Object Constraint Language (OCL). ISO/IEC 19507:2012(E)
13. Nielsen, M., Plotkin, G., Winskel, G.: Petri nets, event structures and domains. *Theor. Comp. Sc.* 1, vol. 13, 85–108 (1981)
14. *Process Algebra for Parallel and Distributed Processing*. Alexander, M., Gardner, W. (eds), CRC Press (2009)
15. UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded Systems. OMG (2011), <http://www.omg.org/spec/MARTE/1.1/pdf/>
16. OMG Unified Modeling LanguageTM (OMG UML), Infrastructure. OMG (2011), <http://www.omg.org/spec/UML/2.4.1/Infrastructure>
17. OMG Unified Modeling LanguageTM (OMG UML), Superstructure. OMG (2011), <http://www.omg.org/spec/UML/2.4.1/Superstructure>

Specializations and Symbolic Modeling

Vladimir Peschanenko¹, Anton Guba² and Constantin Shushpanov³

¹Kherson State University, 27, 40 rokiv Zhovtnya str., Kherson, 73000 Ukraine,

vladimirius@gmail.com

²Glushkov Institute of Cybernetics of NAS of Ukraine, 40, Glushkova ave., Kyiv, 03680,

antonguba@ukr.net

³LLC «Information Software Systems», 15, Bozhenko str., Kyiv, 03680 Ukraine,

costa@iss.org.ua

Abstract. We present the technique that allows splitting first-order logic formulae into parts which helps to use the special algorithms of satisfiability checking and predicate transformer, which are the specializations. We describe the mathematical description of the algorithm of the constructing specializations and a few particular approaches to them, which speed up modeling of industrial models. We prove the correctness of satisfiability and predicate transformer functions. We consider forward and backward applicability of basic protocols during symbolic modeling and verification. We introduce the examples for each specialization. We provide the experiments with typical real examples.

Keywords. Symbolic modeling, satisfiability, predicate transformer

Key terms. FormalMethod, MathematicalModeling, SoftwareComponent, VerificationProcess

1 Introduction

The technique of symbolic verification of requirement specifications of software systems has shown good results in automatic detection of reachability of deadlocks and violation of user-defined properties [1]. In previous works [2-4] symbolic models of systems being transition systems with symbolic states represented by formulae of first order logic were considered. A relation of transitions between the formulae is determined and marked by basic protocols, which are considered as actions, performed in the system. A basic protocol is a formula of dynamic logic $\forall x(\alpha(x, a) \rightarrow \langle P(x, a) \rangle \beta(x, a))$ and it describes local properties of the system in terms of pre- and postconditions α and β . Both are formulae of first order multisorted logic interpreted on a data domain, P is a process, represented by means of MSC dia-

gram and describes the reaction of a system triggered by the precondition, x is a set of typed data variables, and a is a set of environment attributes. The general theory of basic protocols is presented in [5].

A transition is considered as an operator in the space of postcondition formulae. As the operator transforms one formula to another, in [6] a term “predicate transformer” is used. Thus, to compute transitions between the states of such models basic protocols are interpreted as predicate transformers: for given symbolic state of the system and given basic protocol the direct predicate transformer generates the next symbolic state as its strongest postcondition, and the backward predicate transformer generates the previous symbolic state as its weakest precondition. These concepts have been implemented in VRS (Verification of Requirement Specifications) system [7] and IMS (Insertion Modeling System) system [8].

An amount of papers with novel and very efficient techniques for computing satisfiability using SAT/SMT has been published in the last years, and some very efficient SMT tools are now available (e.g., BarceLogic [9], CVCLite/CVC/CVC4 [10,11,12], DLSAT [13], haRVey [14], MathSAT [15], SDSAT [16], TSAT++ [17], UCLID [18], Yices [19], Verifun [20], Zapato [21], Z3 [22]). An amount of benchmarks, mostly derived from verification problems, is available at the SMT-LIB [23]. Workshops devoted to SMT and official competitions on SMT tools are run yearly.

All these tools could be configured with the help of many parameters, which means the usage of some techniques, tactics, heuristics or not, in order to gain in performance. In the paper [24] the algorithm configuration problem is stated as follows: given an algorithm, a set of parameters for the algorithm, and a set of input data, find parameter values under which the algorithm achieves the best possible performance on the input data. It gives a possibility of automated tuning of algorithm for obtaining performance on formulae of some theory.

Usually during modeling of real projects we deal with complex environment states and simple formulae of basic protocols (pre- and postconditions). It means that we should check the satisfiability of the conjunction of the environment state and the precondition formula and transform this whole big formula with the help of predicate transformer [6]. Obviously, the manipulation with whole formulae is not required for most of cases.

For example, let $i, j : \text{int}$, $f : \text{int} \rightarrow \text{int}$ be attributes and $f(i) > 0 \wedge f(0) < 5 \wedge j > 0$ be an environment state, and $1 \rightarrow j := j + 1$ be a basic protocol. Let's apply this basic protocol to the environment state. First, the satisfiability of conjunction of basic protocol precondition and environment state should be checked: $f(i) > 0 \wedge f(0) < 5 \wedge j > 0$. This checking should use the notion of functional symbols: $(i = 0) \rightarrow (f(i) = f(0))$. After that we should apply basic protocol postcondition to conjunction of environment state and precondition (see section Application of Basic Protocol):

$$\begin{aligned} & \exists (v : \text{int})(f(i) > 0 \wedge f(0) < 5 \wedge v > 0 \wedge (j = v + 1)) \Rightarrow \\ & \Rightarrow f(i) > 0 \wedge f(0) < 5 \wedge \exists (v : \text{int})(v > 0 \wedge (j = v + 1)) \Rightarrow \\ & \Rightarrow f(i) > 0 \wedge f(0) < 5 \wedge j > 1 \end{aligned}$$

It is known that basic protocol changes attribute j only (see section about predicate transformers). It means that we could apply basic protocol to small part of environment state that depends on j , but not to whole environment state formula. In this example it could be $j > 0$ only. If there are no predicates in projects, which could compare values of attribute j with values of other attributes, then we could use some special theories for manipulating with such formulae. In this example numeral intervals could be used for representation of values of attribute j . We call such special theories *Specialization of sat, pt functions* according to our general algorithm.

So, the main goal of this paper is to present a mathematical description of algorithm of constructing specializations and a few particular approaches to specialization which speed up modeling of industrial models. This paper is a continuation of the [25], where only concrete values as a kind of specialization were described.

In the Section 2 we describe the process of forward application of a basic protocol with the help of the satisfiability and the forward predicate transformer. In the Section 3 we present an applicability of basic protocols using satisfiability and backward predicate transformer. The specializations by memory usage and functional symbols are proposed in the Section 4. The results of experiments are discussed in the Section 5. In the Section 6 we summarize advantages of usage of the specializations and what could be done in the nearest future.

2 Forward Application of Basic Protocol

Let $S(a)$ be an environment state, $\forall x(\alpha(x, a) \rightarrow \langle P(x, a) \rangle \beta(x, a))$ be a basic protocol, where x – parameters of basic protocol, a – attributes of model, $D(x, a) = E(a) \wedge \alpha(x, a)$ – conjunction of environment state and precondition of basic protocol.

At the first step of application of basic protocol satisfiability of conjunction of environment state and precondition of basic protocol is checked: $sat(D(x, a))$. If the formula is unsatisfiable, then basic protocol is not applicable to environment state $S(a)$. If not, then process $P(x, a)$ is run and after forward predicate transformer is applied: $pt(D(x, a), \beta(x, a))$. The process of $P(x, a)$ is not considered in the paper, because the specialization tries to speed up the functions sat and pt .

2.1 Satisfiability

The checking formula satisfiability function sat is based on the Shostak method, adapted to combination of numerical, symbolic and enumerated data types. If all of the attribute expressions (simple attributes and functional symbols with parameters) that are free in the formula S are simple, then for satisfiability checking it is sufficient to prove validity of the closed formula $\exists(a, x)D(x, a)$, where a is a set of all simple attributes which occur in S , x is a set of parameters of basic protocol. For attribute expressions with parameters (including access functions to the elements of arrays),

the Ackermann reduction of the uninterpreted functional symbols is used, where attribute expression is an attribute or functional symbol with parameters.

The Shostak method consists of the following. An expression of the form $f(x)$ is called as *Functional Expression*, if f is an attribute and x is a list of its parameters. At first, superpositions of functional expressions are eliminated by successive substitution of every internal occurrence of $f(x)$ by a new variable y , bounded by existential quantifier and added to the formula $y = f(x)$. For example, formula $P(f(g(x)))$ is replaced by formula $\exists y(y = g(x) \wedge P(f(y)))$. After all such replacements there will not be complex functional expressions in the formula. Further, for every attribute expression f of functional type all its occurrences $f(x_1), \dots, f(x_n)$ with the different parameters x_1, \dots, x_n are considered. Occurrence $f(x_i)$ is replaced by variable y_i , bounded by existential quantifier and substitutive equations $(x_i = x_j) \rightarrow (y_i = y_j)$ are added. Now in the formula there are only simple attributes, and a method considered in [26] is used.

2.2 Forward Predicate Transformer

In general case, the post-condition looks like $\beta(x, a) = R(x, a) \wedge C(x, a)$, where $R = (r_1 := t_1 \wedge r_2 := t_2 \wedge \dots)$ is a conjunction of assignments and $C(x, a)$ is a formula part of post-condition.

We will consider three sets of functional expressions (we consider attributes as a functional expression with 0 arity): r , s and z . Set $r = (r_1, r_2, \dots)$ consists of the left parts of assignment, and also of other functional expressions that recursively depend on the left parts. In other words, r consists of the left parts of assignments and, if some functional expression f is included into this set, then all functional expressions in which f occurs are also included in r . Set $s = (s_1, s_2, \dots)$ consists of functional expressions which have external occurrences (not in arguments of such functional expressions) in formula part C of post-condition, but do not coincide with expressions from the set r . Finally, set $z = (z_1, z_2, \dots)$ consists of functional expressions which have external occurrences in formula D in right parts of assignments and in internal occurrences (in arguments if functional expressions) of the functional expressions of formula part C of post-condition and left parts of assignments, but these assignments are not included in two other sets (including parameter of basic protocol). Now, considering formulae, from which a post-condition and formula D are constructed as functions of external occurrences of elements of these sets, we get a presentation of post-condition in the following form:

$$B(r, s, z) = (r_1(r, s, z) := t_1(r, s, z) \wedge r_2(r, s, z) := t_2(r, s, z) \wedge \dots) \wedge C(r, s, z),$$

Predicate transformer is determined by the following formula:

$$\text{pt}(D(r, s, z), \beta(r, s, z)) = q_1 \vee q_2 \vee \dots, \text{where}$$

$$q_i = \exists(u, v)(D(u, v, z) \wedge R(u, v, z) \wedge E_i(u, v, z) \wedge C(r, s, z)),$$

$$R(u, v, z) = (r_1(u, v, z) = t_1(u, v, z) \wedge (r_2(u, v, z) = t_2(u, v, z) \wedge \dots)),$$

Formula $R(u, v, z)$ is a quantifier-free part of the assignment formula. Set of the variables $u(v)$ represents new variables for each attribute expression from $r(s)$ set. The pt substitutes attributes from $r(s)$ set to variables from $u(v)$ set in corresponded part of formula.

Each of disjunctive members q_i corresponds to one of possible means of identification of functional expressions occurring in formulae $\beta(x, a)$, and $E_i(u, v, z)$ is a set of equalities and inequalities corresponding to such identification.

To describe the construction of $E_i(u, v, z)$ we will consider the set M of all pairs of functional expressions in the form $(f(k), f(l)), k = (k_1, k_2, \dots), l = (l_1, l_2, \dots)$, where $f(k)$ is chosen from set z , and $f(l)$ – from sets r and s . These functional expressions shall be equal if their arguments were equal before application of basic protocol.

Let's choose arbitrary subset $N \subseteq M$ (including an empty set for every pair $(f(k), f(l)) \in N$ we will consider conjunction of equalities $k = l, (k_1 = l_1 \wedge k_2 = l_2 \wedge \dots)$. We will unite all such conjunctions in one and will add to it conjunctive negations of all equalities, which correspond to pairs which are not included into the set N . We will denote the obtained formula as $G_i(r, s, z)$. If this formula is satisfiable, then the choice is successful. Now obviously, $f(k)$ is not independent and shall change the value because $G_i(r, s, z)$ is true. Thus, $f(k)$ shall change the value in the same way as $f(l)$. Set $E_i(r, s, z) = G_i(r, s, z) \wedge H_i(z, u, v)$ where $H_i(z, u, v)$ is a conjunction of equalities $f(k) = w$ if a variable w corresponds to $f(l)$. Thus, if $f(k)$ coincides with several functional expressions, it is not important what variable is chosen (transitivity of equality) [6].

3 Backward Application of Basic Protocol

Let $S(a)$ be an environment state after the application of the basic protocol $\forall x(\alpha(x, a) \rightarrow \langle P(x, a) \rangle \beta(x, a))$, where x is parameters of basic protocol, a – attributes of model, $\beta(x, a) = R(x, a) \wedge C(x, a)$, where $R = (r_1 := t_1 \wedge r_2 := t_2 \wedge \dots)$ is a conjunction of assignments and C is a formula part of postcondition, $D(x, a) = S(a) \wedge C(x, a)$ is a conjunction of environment state and formula part of postcondition of basic protocol.

3.1 Satisfiability

At first step of application of basic protocol in backward mode satisfiability of conjunction of environment state and formula part of postcondition of basic protocol is checked: $sat(D(x, a))$. If the formula is unsatisfiable, then the basic protocol is not applicable to environment state $S(a)$. If not, then process $P(x, a)$ is run and after a backward predicate transformer is applied: $pt^{-1}(D(x, a), \beta(x, a))$.

3.2 Backward Predicate Transformer

A backward predicate transformer considers three sets of functional expressions r , s and z (as forward too). A postcondition of the basic protocols is represented by the following formula:

$$B(r, s, z) = (r_1(r, s, z) := t_1(r, s, z) \wedge r_2(r, s, z) := t_2(r, s, z) \wedge \dots) \wedge C(r, s, z)$$

A backward predicate transformer is determined by the following formula:

$$pt^{-1}(D(r, s, z), \beta(r, s, z)) = q_1^{-1} \vee q_2^{-1} \vee \dots, \text{ where}$$

$$q_i^{-1} = \exists(u, v)(D(u, v, z) \wedge R'(u, r, s, z) \wedge E_i(u, v, z)) \wedge \alpha(r, s, z),$$

$$R'(u, r, s, z) = (u_1(r, s, z) = t_1(r, s, z)) \wedge (u_2 = t_2(r, s, z)) \wedge \dots, u = \{u_1, u_2, \dots\},$$

Each of disjunctive members q_i corresponds to one of possible identification of functional expressions, occurring in formulae $\beta(x, a)$ and environment state $S(a)$, where $E_i(u, v, z)$ are sets of equalities and inequalities corresponding to such identification. Formula $E_i(u, v, z)$ is built in the same way as in forward predicate transformer [27].

4 Specialization

We propose to use two types of specializations:

1. Specialization by memory usage
2. Specialization by functional symbol

4.1 Specialization by memory usage

Let a_1, a_2 be sets of attributes from initial environment state and $a_1 \cap a_2 = \emptyset \wedge a_1 \cup a_2 = a$, $S(a) = S_1(a_1) \wedge S_2(a_2)$ is environment state, $B(x, a) = \forall x(\alpha_1(x_1, a_1) \wedge \alpha_2(x_2, a_2) \rightarrow \langle P(x, a) \rangle \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2))$ is basic protocol, where $x_1 \cap x_2 = \emptyset \wedge x_1 \cup x_2 = x$.

If $B'(x, a) = \forall x(\bigvee_i \alpha_i(x, a) \rightarrow \langle P(x, a) \rangle \beta(x, a))$ then $sat(S(a) \wedge (\bigvee_i \alpha_i(x, a))) = \bigvee_i sat(S(a) \wedge \alpha_i(x, a))$ and $pt(S(a) \wedge (\bigvee_i \alpha_i(x, a)), \beta(x, a)) = \bigvee_i pt(S(a) \wedge \alpha_i(x, a), \beta(x, a))$. So, in the next text we consider basic protocol as $B(x, a)$ only.

4.2 Theorem 1

$$\begin{aligned} & sat(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2)) = \\ & = sat(S_1(a_1) \wedge \alpha_1(x_1, a_1)) \wedge sat(S_2(a_2) \wedge \alpha_2(x_2, a_2)) \end{aligned}$$

Proving.

Function sat builds closed formula. So,

$$\begin{aligned} & sat(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2)) = \\ & = \exists(v_1, v_2, x_1, x_2)(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2)) \end{aligned}$$

where v_1, v_2 are variables generated for attribute expression which depend on attributes a_1, a_2 . It is known that $a_1 \cap a_2 = \emptyset \wedge a_1 \cup a_2 = a \wedge x_1 \cap x_2 = \emptyset \wedge x_1 \cup x_2 = x$. It means that scope of quantifiers could be narrowed:

$$\begin{aligned} & \exists(v_1, v_2, x_1, x_2)(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2)) = \\ & = \exists(v_1, x_1)(S_1(a_1) \wedge \alpha_1(x_1, a_1)) \wedge \exists(v_2, x_2)(S_2(a_2) \wedge \alpha_2(x_2, a_2)) = i \\ & = sat(S_1(a_1) \wedge \alpha_1(x_1, a_1)) \wedge sat(S_2(a_2) \wedge \alpha_2(x_2, a_2)) \end{aligned}$$

Theorem is proved.

This theorem means the following:

1. If $S(a) = S_1(a_1) \wedge S_2(a_2)$ and $\alpha(a, x) = \alpha_1(x_1, a_1)$ and $S(a)$ is satisfiable, then it is enough to check satisfiability of conjunction of $S_1(a_1) \wedge \alpha_1(x_1, a_1)$ for satisfiability checking of $S(a) \wedge \alpha(x, a)$. Checking of satisfiability of $S_2(a_2)$ is not required.
2. Checking of each part $sat(S_i(a_i) \wedge \alpha_i(x_i, a_i))$ could be done concurrently.

This case could be easily generalized to a_1, \dots, a_n case, because if it is possible to build subsets $a_i^1, a_i^2 \in a_i \wedge a_i^1 \cap a_i^2 = \emptyset \wedge a_i^1 \cup a_i^2 = a_i$ and to split an environment state and basic protocol accordingly to the *theorem 1*, then $sat(\bigwedge_i S_i(a_i) \wedge \alpha_i(x_i, a_i)) = \bigwedge_i sat(S_i(a_i) \wedge \alpha_i(x_i, a_i))$. So, after if we say about such pair of two sets $a_i^1, a_i^2 \in a_i \wedge a_i^1 \cap a_i^2 = \emptyset \wedge a_i^1 \cup a_i^2 = a_i$, then we understand that it could be applicable and for n sets.

Let's see how forward and backward predicate transformer can be applied.

4.3 Theorem 2

For forward application of basic protocol it is true that:

$$\begin{aligned} & pt(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2), \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2)) = \\ & = pt(S_1(a_1) \wedge \alpha_1(x_1, a_1), \beta_1(x_1, a_1)) \wedge pt(S_2(a_2) \wedge \alpha_2(x_2, a_2), \beta_2(x_2, a_2)) \end{aligned}$$

Proving.

pt function builds sets r, s, z from postcondition $\beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2)$ and formula $S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2)$, where r is a set of attribute expressions from left parts of assignments of postcondition, s is a set of attribute expressions from formula part of postcondition, z is a set of other attribute expressions from formula and postcondition. We know that sets of attribute expressions from pairs $S_1(a_1) \wedge \alpha_1(x_1, a_1)$, $\beta_1(x_1, a_1)$ and $S_2(a_2) \wedge \alpha_2(x_2, a_2)$, $\beta_2(x_2, a_2)$ are not intersected. It means that we could split each set r, s, z on subsets

$r = r_1 \cup r_2, s = s_1 \cup s_2, z = z_1 \cup z_2$ and $r_1 \cap r_2 = \emptyset, s_1 \cap s_2 = \emptyset, z_1 \cap z_2 = \emptyset$, because $a_1 \cap a_2 = \emptyset$. Let's write formula which is built by *pt* function.

Let $D(a, x) = D_1(x_1, a_1) \wedge D_2(x_2, a_2), D_1(x_1, a_1) = S_1(a_1) \wedge \alpha_1(x_1, a_1)$,
 $D_2 = S_2(a_2) \wedge \alpha_2(x_2, a_2)$ and $\beta_1(x_1, a_1) = R_1(r_1, s_1, z_1) \vee C_1(r_1, s_1, z_1)$,
 $\beta_2(x_2, a_2) = R_2(r_2, s_2, z_2) \vee C_2(r_2, s_2, z_2)$. So, general formula of predicate transformer is the following:

$$\bigvee_i q_i = \exists(u, v)(D(u, v, z) \wedge R(u, v, z) \wedge (\bigvee_i E_i(u, v, z)) \wedge C(r, s, z))$$

where $R(u, v, z) = R_1(u_1, v_1, z_1) \wedge R_2(u_2, v_2, z_2)$, $C(r, s, z) = C_1(r_1, s_1, z_1) \wedge C_2(r_2, s_2, z_2)$. because $\beta(a, x) = \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2)$.

Let's write in details how to obtain $\bigvee_i E_i(u, v, z)$. It is known that $r_1 \cap r_2 = \emptyset, s_1 \cap s_2 = \emptyset, z_1 \cap z_2 = \emptyset$. To build such disjunction we should take into account all pairs of functional attribute expressions from sets r, s and z . It means that each such pair should be in set of attribute $(r_1 \wedge s_1; z_1)$ or $(r_2 \wedge s_2; z_2)$. So,
 $\bigvee_i E_i(u, v, z) = (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1)) \wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2))$

Let's consider formula of predicate transformer:

$$\begin{aligned} & pt(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2), \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2)) = \\ & \bigvee_i q_i = \exists(u, v)(D(u, v, z) \wedge R(u, v, z) \wedge (\bigvee_i E_i(u, v, z)) \wedge C(r, s, z)) = \\ & = \exists(u_1, u_2, v_1, v_2)(D_1(u_1, v_1, z_1) \wedge D_2(u_2, v_2, z_2) \wedge \\ & \wedge R_1(u_1, v_1, z_1) \wedge R_2(u_2, v_2, z_2) \wedge \\ & \wedge (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1)) \wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2)) \wedge \\ & \wedge C_1(r_1, s_1, z_1) \wedge C_2(r_2, s_2, z_2)) = \\ & = \exists(u_1, v_1)(D_1(u_1, v_1, z_1) \wedge R(u_1, v_1, z_1) \wedge (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1)) \wedge \\ & \wedge C_1(r_1, s_1, z_1)) \wedge \exists(u_2, v_2)(D_2(u_2, v_2, z_2) \wedge R(u_2, v_2, z_2) \wedge \\ & \wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2)) \wedge C_2(r_2, s_2, z_2)) \wedge \dots = \\ & = pt(D_1(x_1, a_1), \beta_1(x_1, a_1)) \wedge pt(D_2(x_2, a_2), \beta_2(x_2, a_2)) \end{aligned}$$

Theorem is proved.

4.4 Theorem 3

For backward mode it is true that:

$$\begin{aligned}
& pt^{-1}(S_1(a_1) \wedge C_1(r_1, s_1, z_1) \wedge S_2(a_2) \wedge C_2(r_2, s_2, z_2), \\
& \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2)) = pt^{-1}(S_1(a_1) \wedge C_1(r_1, s_1, z_1), \beta_1(x_1, a_1)) \wedge \\
& pt(S_2(a_2) \wedge C_2(r_2, s_2, z_2), \beta_2(x_2, a_2))
\end{aligned}$$

Proving.

$$\begin{aligned}
R(u, v, z) &= R(u_1, v_1, z_1) \wedge R(u_2, v_2, z_2), \quad C(r, s, z) = C_1(r_1, s_1, z_1) \wedge C_2(r_2, s_2, z_2) \\
\text{because } \beta(r, x) &= \beta_1(a_1, x_1) \wedge \beta_2(a_2, x_2). \quad \bigvee_i E_i(u, v, z) = (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1)) \wedge \\
&\wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2)) \text{ from previous theorem.}
\end{aligned}$$

$$\begin{aligned}
pt^{-1}(S(a) \wedge C(r, s, z), \beta(r, s, z)) &= \bigvee_i q_i^{-1} = \\
&= \bigvee_i \exists(u, v)(S(a) \wedge C(r, s, z) \wedge R'(u, r, s, z) \wedge E_i(u, v, z)) \wedge \alpha(r, s, z) = \\
&= \exists(u_1, u_2, v_1, v_2)(S_1(r_1, s_1, z_1) \wedge S_2(r_2, s_2, z_2) \wedge \\
&\wedge C_1(r_1, s_1, z_1) \wedge C_2(r_2, s_2, z_2) \wedge \\
&\wedge (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1)) \wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2))) \wedge \\
&\wedge \alpha_1(r_1, s_1, z_1) \wedge \alpha_2(r_2, s_2, z_2) = \\
&= \exists(u_1, v_1)(S_1(r_1, s_1, z_1) \wedge C_1(r_1, s_1, z_1) \wedge (\bigvee_{i_1} E_{i_1}(u_1, v_1, z_1))) \wedge \alpha_1(r_1, s_1, z_1) \wedge \\
&\wedge \exists(u_2, v_2)(S_2(r_2, s_2, z_2) \wedge C_2(r_2, s_2, z_2) \wedge (\bigvee_{i_2} E_{i_2}(u_2, v_2, z_2))) \wedge \alpha_2(r_2, s_2, z_2) = \\
&= pt^{-1}(S_1(a_1) \wedge C_1(r_1, s_1, z_1), \beta_1(x_1, a_1)) \wedge pt(S_2(a_2) \wedge C_2(r_2, s_2, z_2), \beta_2(x_2, a_2))
\end{aligned}$$

Theorem is proved.

Theorem 2 and theorem 3 mean that:

1. Functions pt , pt^l could be applied separately and concurrently.
2. If postcondition contains $\beta_1(x_1, a_1)$ only, then

$$\begin{aligned}
& pt(S_1(a_1) \wedge \alpha_1(x_1, a_1) \wedge S_2(a_2) \wedge \alpha_2(x_2, a_2), \beta_1(x_1, a_1)) = \\
& = S_2(a_2) \wedge \alpha_2(x_2, a_2) \wedge pt(S_1(a_1) \wedge \alpha_1(x_1, a_1), \beta_1(x_1, a_1)) \\
& pt^{-1}(S_1(a_1) \wedge C_1(r_1, s_1, z_1) \wedge S_2(a_2) \wedge C_2(r_2, s_2, z_2), \beta_1(x_1, a_1)) = \\
& = S_2(a_2) \wedge C_2(r_2, s_2, z_2) \wedge pt^{-1}(S_1(a_1) \wedge C_1(r_1, s_1, z_1), \beta_1(x_1, a_1))
\end{aligned}$$

So, functions $sat(D_i(x_i, a_i))$, $pt(D_i(x_i, a_i), \beta_i(x_i, a_i))$ are called specialization, because we could use some special theories for implementation of it.

5 Examples of Usage of Specializations

5.1 Examples of Specializations by Memory Usage

Example 1. Concrete values. Let $S(a) = (i = 2) \wedge S(a/i)$ be an environment state where $i : \text{int}$ and a/i is a set of all attributes in model except i , $b = \forall x((i > 0) \rightarrow \langle \rangle (i := i + 1))$. For application of such basic protocol we should check satisfiability of the next formula: $\text{sat}((i = 2) \wedge (i > 0)) = 1$, and the postcondition should be applied to $(i = 2) : \exists v((v = 2) \wedge (v > 0) \wedge (i = v + 1)) \Rightarrow (i = 3)$. For such examples direct C++ translation could be used instead of using some special theories, and it will work much faster because it doesn't require any additional checking, just direct translation into C++ code and compilation of it.

Example 2. Let $S(a) = (i < 2) \wedge S(a/i)$ be environment state where $i : \text{int}$ and a/i is a set of all attributes in model except i , $b = \forall x((i > 0) \rightarrow \langle \rangle (i := i + 1))$. For application of such basic protocol we should check satisfiability of the next formula: $\text{sat}((i < 2) \wedge (i > 0)) = 1$, and the postcondition should be applied to $(i < 2) : \exists v((v < 2) \wedge (v > 0) \wedge (i = v + 1)) \Rightarrow (i > 1) \wedge (i < 3)$. For such examples numerical intervals could be used. So, $S(a) = (i \in (-\infty; 2)) \wedge S(a/i)$, $b = \forall x((i \in (0; +\infty)) \rightarrow \langle \rangle (i := i + 1))$. Satisfiability checking looks like just crossing of two numerical intervals: $i \in (-\infty; 2) \cap (0; +\infty) \Rightarrow i \in (0; 2) \Rightarrow i \in [1; 1]$ for integer. Application of pt creates the following formula: $\exists v((v \in [1; 1]) \wedge (i = v + 1)) \Rightarrow \Rightarrow i - 1 \in [1; 1] \Rightarrow i \in [2; 2]$. This approach will work faster than general satisfiability checking and quantifiers eliminations. Such approach could be used for all numeric and enumerated types.

5.2 Examples of Specializations by Functional Symbol

It is not always possible to represent environment state and basic protocols in the following way: $S(a) = S_1(a_1) \wedge S_2(a_2)$, and $B(a, x) = \forall x(\alpha_1(x_1, a_1) \wedge \alpha_2(x_2, a_2) \rightarrow \langle P(x, a) \rangle \beta_1(x_1, a_1) \wedge \beta_2(x_2, a_2))$ where $a_1 \cap a_2 = \emptyset$, $a_1 \cup a_2 = a$, $x_1 \cap x_2 = \emptyset \wedge x_1 \cup x_2 = x$. One of such situation occurs when a value of functional attribute expression and its parameter has different types and belongs to the different subsets a_i . For example, if functional attribute: $i, j : \text{int}, f : \text{int} \rightarrow T$ is defined where $T \in (c_1, c_2, c_3)$ is enumerated type with three enumerated constants: c_1, c_2, c_3 , then formula $(f(i) = c_1) \wedge i > 0$ could be represented with specializations as follows: $(f : v_1 = f(i)) \wedge (v_1 = c_1) \wedge (i > 0)$. Let $b = 1 \rightarrow \langle \rangle (f(j) := c_2)$ be a basic protocol. Its specialized representation is: $b = 1 \rightarrow \langle \rangle (f : v_1 = f(j)) \wedge (v_1 := c_2) \wedge 1$. It is required to merge such data structures for pt function which should consider all pairs of functional attribute expression from sets r, s and z :

$(f : v_1 = f(i)) \wedge (f : v_1 = f(j)) \Rightarrow (f : v_1 = f(i), v_2 = f(j))$. After that basic protocol should be transformed in the following form: $b = 1 \rightarrow \langle (f : v_2 = f(j)) \wedge (v_2 := c_2) \wedge 1$. It is required to take into account two possible combinations: $(i = j) \vee \neg(i = j)$. So, we obtain:

$$\begin{aligned}
& pt((f : v_1 = f(i), v_2 = f(j)) \wedge (v_1 = c_1) \wedge i > 0, v_2 := c_2) = \\
& = (f : v_1 = f(i), v_2 = f(j)) \wedge \\
& \wedge \exists v((i = j) \wedge (v = c_1) \wedge (v_2 = c_2)) \wedge \\
& \wedge \exists v(\neg(i = j) \wedge (v_1 = c_1) \wedge (v_2 = c_2)) \wedge i > 0 \Rightarrow \\
& \Rightarrow (f : v_1 = f(i), v_2 = f(j)) \wedge (v_2 = c_2) \wedge i > 0 \wedge (i = j) \vee \\
& \vee (f : v_1 = f(i), v_2 = f(j)) \wedge (v_1 = c_1) \wedge (v_2 = c_2) \wedge i > 0 \wedge \neg(i = j) \Rightarrow \\
& \Rightarrow (f : v_1 = f(i)) \wedge (v_1 = c_2) \wedge i > 0 \wedge (i = j) \vee \\
& \vee (f : v_1 = f(i), v_2 = f(j)) \wedge (v_1 = c_1) \wedge (v_2 = c_2) \wedge i > 0 \wedge \neg(i = j)
\end{aligned}$$

Let $S(a) = F(f_1, f_2, \dots, v_1, v_2, a_1, a_2) \wedge S_1(a_1) \wedge S_2(a_2)$ be an environment state where $f_1 \neq f_2 \neq \dots$ are names of functional symbols, v_1, v_2 are variables for each functional attribute expression from sets a_1, a_2 correspondently, and

$$\begin{aligned}
& F(f_1, f_2, \dots, v_1, v_2, a_1, a_2) = \\
& = (f_1 : v_1^1 = f_1(t_1^1, t_1^2, \dots), v_1^2 = f_1(t_1^1, t_1^2, \dots), \dots, \\
& f_2 : v_2^1 = f_2(t_2^1, t_2^2, \dots), v_2^2 = f_2^2(t_2^1, t_2^2, \dots), \dots, \dots)
\end{aligned}$$

where $v_1^1, v_1^2 \in a_{f_1}, v_2^1, v_2^2 \in a_{f_2}, \dots$ are variables of type of functional names f_1, f_2, \dots for each attribute expression, a_{f_i} is set of attribute, such as $f_i \in a_j$, $t_i^j \in a_i^i \in \{a_i\}$, ... - corresponded arguments for each functional with the same name are in one specialization, and Shostak's method could be applied for each right part of equation in F .

Let $S(a) = F(f_1, f_2, \dots, v_1, v_2, a_1, a_2) \wedge S_1(a_1) \wedge S_2(a_2)$. and
 $b(a) = \forall x (F_b(f_1, f_2, \dots, v_1, v_2, a_1, a_2, x_1, x_2) \wedge \alpha_1(v_1, x_1, a_1) \wedge$
 $\wedge \alpha_2(v_2, x_2, a_2) \rightarrow \langle P(a, x) \rangle \beta_1(v_1, x_1, a_1) \wedge \beta_2(v_2, x_2, a_2))$

5.3 Theorem 4

$$\begin{aligned}
& sat(S(a) \wedge \alpha(x, a)) = sat(\bigwedge_{(i,k,l)} ((f_i(t_i^1, t_i^2, \dots) = f_i(t_i^1, t_i^2, \dots)) \rightarrow (v_i^k = v_i^l)) \wedge \\
& \wedge S_1(v_1', a_1) \wedge \alpha_1(v_1', x_1, a_1) \wedge S_2(v_2', a_2) \wedge \alpha_2(v_2', x_2, a_2)) = \\
& = \bigvee_i sat(q_i \wedge S_i(v_i', a_i) \wedge \alpha_i(v_i', a_i, x_i))
\end{aligned}$$

where $f_i(t_i^1, t_i^2, \dots) = f_i(t_i^1, t_i^2, \dots)$ is equality of arguments of functional attribute expressions.

Proving

Let's define $F'(f_1, f_2, \dots, v'_1, v'_2, a_1, a_2, x_1, x_2) = F(f_1, f_2, \dots, v_1^1, v_2^1, a_1, a_2) \cup F_b(f_1, f_2, \dots, v_1^2, v_2^2, a_1, a_2, x_1, x_2)$. We combine all equations with the same name of functional symbol f_i and renaming variables names after such union for equations from basic protocol. After that we obtain sets of variables v'_1, v'_2 and new basic proto-

col $b(a) = \forall x (F_b(f_1, f_2, \dots, v'_1, v'_2, a_1, a_2, x_1, x_2) \wedge \alpha_1(v'_1, x_1, a_1) \wedge \alpha_2(v'_2, x_2, a_2) \wedge \dots \rightarrow \langle P(a, x) \rangle \alpha_1(v'_1, x_1, a_1) \wedge \alpha_2(v'_2, x_2, a_2))$.

For satisfiability checking we should add corresponded implication for each pair of equation from $F'(f_1, f_2, \dots, v'_1, v'_2, a_1, a_2, x_1, x_2)$ with the same name of functional symbol f_i .

$$\begin{aligned} & \bigwedge_{(i,k,l)} ((f_i(t_i^1, t_i^2, \dots) = f_i^l(t_i^1, t_i^2, \dots)) \rightarrow (v_i^k = v_i^l)) = \\ & = \bigwedge_{i,k,l} (\neg(f_i(t_i^1, t_i^2, \dots) = f_i^l(t_i^1, t_i^2, \dots)) \vee (v_i^k = v_i^l)) \end{aligned}$$

Each left and right parts of equation and negation of equations are in the same specialization. It means that we could build here a disjunction of conjunction. Each conjunct in such disjunction is q_i which will be in one form of our specialization. So, it means that we could check satisfiability in the following form $\bigvee_i \text{sat}(q_i \wedge S_i(v'_i, a_i) \wedge \alpha'_i(v'_i, a_i, x_i))$.

Theorem is proved.

5.4 Theorem 5

$$\begin{aligned} & \text{pt}(S(a) \wedge \alpha(x, a), \beta(x, a)) = \\ & = \bigvee_i (\text{pt}'(E_1^i(v'_1, x_1, a_1), S_1(v'_1, a_1) \wedge \alpha_1(v'_1, x_1, a_1), \beta_1(x_1, a_1))) \wedge \\ & \wedge \text{pt}'(E_2^i(v'_2, x_2, a_2), S_2(v'_2, a_2) \wedge \alpha_2(v'_2, x_2, a_2), \beta_2(x_2, a_2)))) \end{aligned}$$

where

$$\begin{aligned} & \text{pt}'(E_j^i(v'_j, x_j, a_j), S_j(v'_j, a_j) \wedge \alpha_j(v'_j, x_j, a_j), \beta_j(x_j, a_j)) = \bigvee_k q'_k, \\ & q'_k = \exists(u, v) (S_i(u, v, z) \wedge \alpha_i(u, v, z) \wedge R(u, v, z) \wedge \\ & \wedge E_i^j(u, v, z) \wedge E_k^j(u, v, z) \wedge C(r, s, z)) \end{aligned}$$

Proving.

The sets v'_1, v'_2 are built in the same way as in *theorem 3*. Let's consider a general formula of predicate transformer:

$$\text{pt}(D(r, s, z), \beta(r, s, z)) = \bigvee_i \exists(u, v) (D(u, v, z) \wedge R(u, v, z) \wedge E_i(u, v, z) \wedge C(r, s, z)).$$

Coefficient $\bigvee_i E_i(u, v, z)$ looks like disjunction of conjunction of all possible matchings with functional attribute expressions from sets r, s and z . So, we can present

it as conjunction of two disjunctions: $\bigvee_i E_i(u, v, z) = (\bigvee_k E_k(u, v, z)) \wedge (\bigvee_l E_l(u, v, z))$ where $\bigvee_k E_k(u, v, z)$ is disjunction for matching of functional attribute expression where parameters and its value are from different sets of a_j . $\bigvee_l E_l(u, v, z)$ is a disjunction of matching of other functional attribute expression. Each conjunct of such disjunction could be considered as a conjunction which depends on different sets of memory a_j . It means that disjunction of conjunction $\bigvee_k E_k(u, v, z)$ could be prepared early before calling of some pt function without corresponded substitution of x, y . So, $\bigvee_k E_k(u, v, z) = \bigvee_k E_1^k(v'_1, x_1, a_1) \wedge E_2^k(v'_2, x_2, a_2)$. Disjunction $\bigvee_l E_l(u, v, z)$ could be presented in the same way. So, the theorem is proved.

5.5 Theorem 6

$$\begin{aligned} & pt^{-1}(S(a) \wedge C(r, s, z), \beta(x, a)) = \\ & = \bigvee_i (pt'^{-1}(E_1^i(v'_1, x_1, a_1), S_1(v'_1, a_1) \wedge C_1(v'_1, x_1, a_1), \beta_1(x_1, a_1))) \wedge \\ & \wedge pt'^{-1}(E_2^i(v'_2, x_2, a_2), S_2(v'_2, a_2) \wedge C_2(v'_2, x_2, a_2), \beta_2(x_2, a_2)))) \end{aligned}$$

where

$$\begin{aligned} & pt'^{-1}(E_j^i(v'_j, x_j, a_j), S_j(v'_j, a_j) \wedge C_j(v'_j, x_j, a_j), \beta_j(x_j, a_j)) = \bigvee_k q'_k, \\ & q'_k \exists (u, v) (S_j(v'_j, a_j) \wedge C_j(v'_j, x_j, a_j) \wedge R'(u, r, s, z) \wedge \\ & \wedge E_j^j(u, v, z) \wedge E_k(u, v, z) \wedge \alpha_j(r, s, z)) \end{aligned}$$

This theorem could be proved in the same mode as theorem 4.

6 Experiments

In this section we present some results from our test suites. All experiments are divided into several groups. We compare the time of modeling of the satisfiability and the predicate transformer, presented in the Section 2, and these algorithms with the specializations.

The first group of experiments refers to specialization by memory usage. Projects contain formulae in which some attributes have only concrete values. Let us present one typical real example. This example has a functional attribute of symbolic type with integer parameters, simple enumerated and simple integer attributes. All of these integer attributes initialize with concrete values and have concrete values at all times during trace generation (basic protocols do not change those to symbolic ones). Other attributes are symbolic. We provide a specialization for attributes, which are always concrete. The difference of modeling time for this example and for this one specialized by concrete values is more than in 3 times. Of course, the speedup depends on

project: more concrete attributes we have, more speedup we shall obtain. In [25] it was shown that speedup could be in thousands times.

The second group of experiments refers also to specialization by memory usage, but not to concrete values. Examples from this group have enumerated attributes and integer attributes. Some of the integer attributes memories are intersected, some of them are independent. First of all, we provide the splitting of formulae into two parts according to attribute types: enumerated part and integer part. For the enumerated part we use *bitsets*, for integer – common Pressburger algorithm. Speedup was about 5-7%. After we specialize an integer part. We consider the attributes which memory is independent and obtain speedup in 10 times.

So, the results of comparison of modeling time using general satisfiability functions and functions with specialization are given.

Table 6. Results of experiments

Group of tests	General algorithm	With specializations
1	930 sec	300 sec (memory usage/concrete values)
2	300 sec	280 sec (splitting by types)
3	300 sec	33 sec (memory usage/independent memory)

7 Conclusions

Symbolic modeling is a powerful technique for the automated reachability of deadlocks and violations of user-defined properties. The main complexity of the reachability problem is in the complexity of satisfiability and predicate transformer functions. There are a lot of SMT-based techniques which speed up the satisfiability of formulae that satisfy some particular theory. We propose a technique that allows to speedup classical symbolic modeling when formulae could be splitted in several parts and used some special theories for manipulations with them, which are called specializations. The mathematical description of the algorithm for constructing specializations is provided and the correctness of such specializations is proved.

Specializations by memory usage and functional symbols are considered and examples for each are given.

The nearest plans are the investigation of additional kinds of specialization, because the more specializations we have, the more speedup we obtain.

References

1. Symbolic Modeling, http://en.wikipedia.org/wiki/Model_checking
2. Letichevsky, A., Gilbert, D.: A Model for Interaction of Agents and Environments. In: Bert, D., Choppy, C., Moses, P. (eds.) Recent Trends in Algebraic Development Techniques. LNCS 1827, pp. 311–328. Springer Verlag, Berlin Heidelberg (1999)

3. Letichevsky, A.: Algebra of Behavior Transformations and its Applications. In: Kudryavtsev, V. B., Rosenberg, I. G. (eds.) *Structural Theory of Automata, Semigroups, and Universal Algebra*, NATO Science Series II. Mathematics, Physics and Chemistry, vol. 207, pp. 241–272. Springer Verlag, Berlin Heidelberg (2005)
4. Letichevsky A., Kapitonova J., Kotlyarov V., Letichevsky Jr., A., Nikitchenko N., Volkov, V., Weigert T.: Insertion Modeling in Distributed System Design. *Problems of Programming*, (4), 13–39 (2008)
5. Letichevsky, A., Kapitonova, J., Volkov, V., Letichevsky Jr., A., Baranov, S., Kotlyarov, V., Weigert, T.: System Specification with Basic Protocols. *Cybernetics and System Analysis*, (4), 3–21 (2005)
6. Letichevsky, A. A., Godlevsky, A. B., Letichevsky Jr., A. A., Potienko, S. V., Peschanenko, V. S.: Properties of Predicate Transformer of VRS System. *Cybernetics and System Analyses*, (4), 3–16 (2010)
7. Letichevsky, A., Kapitonova, J., Letichevsky Jr., A., Volkov, V., Baranov, S., Kotlyarov, V., Weigert, T.: Basic Protocols, Message Sequence Charts, and the Verification of Requirements Specifications, In: *ISSRE 2004, WITUL (Workshop on Integrated reliability with Telecommunications and UML Languages)*, Rennes, 4 November (2005)
8. Letichevsky, A., Letychevskiy, O., Peschanenko, V.: Insertion Modeling System. In: Clarke, E.M., Virbitskaite, I., Voronkov, A. (eds.) *PSI 2011. LNCS 7162*, pp. 262–274, Springer Verlag, Berlin Heidelberg (2011)
9. Bofill, M., Nieuwenhuis, R., Oliveras, A., Rodríguez-Carbonell, E., Rubio, A.: The Barcelogic SMT Solver. In: Gupta, Aarti and Malik, Sharad (eds.) *CAV 2008. LNCS 5123*, pp. 294–298, Springer Verlag, Berlin Heidelberg (2008)
10. Barrett, C., Berezin, S.: CVC Lite: A New Implementation of the Cooperating Validity Checker. In: Rajeev, A., Peled, D.A. (eds.) *CAV '04. LNCS 3114*, pp. 515–518, Springer Verlag, Berlin Heidelberg (2004)
11. Barrett, C., Tinelli, C.: CVC3. In: W. Damm and H. Hermanns (eds.) *CAV '07. LNCS 4590*, pp. 298–302, Springer Verlag, Berlin Heidelberg (2007)
12. Barrett, C., Conway, C. L., Deters, M., Hadarean, L., Jovanović, D., King, T., Reynolds, A., Tinelli, C.: CVC4. In: Gopalakrishnan, G., Qadeer, S. (eds.) *CAV'11. LNCS 6806*, pp. 171–177, Springer Verlag, Berlin Heidelberg (2011)
13. Cotton, S., Asarin, E., Maler, O., Niebert, P.: Some Progress in Satisfiability Checking for Difference Logic. In: *Proc. FORMATS-FTRTFT (2004)*
14. Déharbe, D., Ranise, S.: Bdd-Driven First-Order Satisfiability Procedures (extended version). Research report 4630, LORIA (2002)
15. Bozzano, M., Bruttomesso, R., Cimatti, A., Junttila, T., van Rossum, P., Schulz, S., Sebastiani, R.: An Incremental and Layered Procedure for the Satisfiability of Linear Arithmetic Logic. In: Halbwachs, Lenore (eds.) *TACAS'05. LNCS 3440*, pp. 317–333, Springer Verlag, Berlin Heidelberg (2005)
16. Ganai, M. K., Talupur, M., Gupta, A.: SDSAT: Tight Integration of Small Domain Encoding and Lazy Approaches in a Separation Logic Solver. In: H. Hermanns, J. Palsberg. (eds.) *TACAS 2006. LNCS 3920*, pp. 135–150. Springer Verlag, Berlin Heidelberg (2006)
17. Audemard, G., Bertoli, P. G., Cimatti, A., Kornilowicz, A., Sebastiani, R.: A SAT based Approach for Solving Formulas over Boolean and Linear Mathematical Propositions. In: A. Voronkov (ed.) *CADE 2002. LNCS (LNAI) 2392*, pp. 195–210. Springer Verlag, Berlin Heidelberg (2002)
18. Bryant, R.E., Lahiri, S.K., Seshia, S.A.: Modeling and Verifying Systems using a Logic of Counter Arithmetic with Lambda Expressions and Uninterpreted Functions.. In: Brinksma

- K., Larsen G. (eds) CAV'04. LNCS 2404, pp. 78–92, Springer Verlag, Berlin Heidelberg (2002)
19. Dutertre, B., de Moura, L.: A Fast Linear-Arithmetic Solver for DPLL(T). In T. Ball and R.B. Jones, (eds.) CAV'06. LNCS 4144, pp. 81–94, Springer Verlag, Berlin Heidelberg (2006)
 20. Walther, C., Schweitzer, S.: About veriFun. In: F. Baader (eds.) CADE'03. LNCS 2741, pp. 322–327, Springer Verlag, Berlin Heidelberg (2003)
 21. Ball, T., Cook, B., Lahiri, S.K., Zhang, L.: Zapato: Automatic Theorem Proving for Predicate Abstraction Refinement. In: Alur, R. A., Peled D. A. (eds.) CAV'04. LNCS 3114, pp. 457–461. Springer Verlag, Berlin Heidelberg (2004)
 22. de Moura, L., Bjørner, N.: Z3: An Efficient SMT Solver. In: C. R. Ramakrishnan, J. Rehof (eds.) TACAS'08, LNCS 4963, pp. 337–340. Springer Verlag, Berlin Heidelberg (2004)
 23. Barrett, C., de Moura, L., Ranise, S., Stump, A., Tinelli, C.: The SMT-LIB Initiative and the Rise of SMT. In: Barner S., Harris I. (eds.) HVC 2010. LNCS 6504, pp. 3–3, Springer Verlag, Berlin Heidelberg (2010)
 24. Hutter, F., Hoos, H.H., Leyton-Brown, K., Stuetzle, T.: ParamILS: an Automatic Algorithm Configuration Framework. JAIR, 36, 267–306 (2009)
 25. Peschanenko, V. S., Guba, A. A., Shushpanov, C. I.: Mixed Concrete-Symbolic Predicate Transformer. Bulletin of Taras Shevchenko National University of Kyiv, Series Physics & Mathematics, 2 (2013) (in press)
 26. Barrett, C., Sebastiani, R., Seshia, S., Tinelli, C.: Satisfiability Modulo Theories. Frontiers in Artificial Intelligence and Applications, 185, 825–885 (2009)
 27. Godlevsky, A. B.: Predicate Transformers in the Context of Symbolic Modeling of Transition Systems. Cybernetics and System Analysis, 4, 91–99 (2010)

On a Dynamic Logic for Graph Rewriting [★]

Mathias Winckel and Ralph Matthes

Institut de Recherche en Informatique de Toulouse (IRIT)

{winckel, matthes}@irit.fr

Abstract. Initially introduced by P. BALBIANI, R. ECHAHED and A. HERZIG, this dynamic logic is useful to talk about properties on term-graphs and to characterize transformations on these graphs. Also are presented the deterministic labelled graphs for which the logical framework is designed.

This logic has been the starting point of a formal development, using the Coq proof assistant, to design a logical and algorithmic framework useful for verifying and proving graph rewriting. The formalization allowed us to figure out some ambiguities in the involved concepts. This formalization is not the topic here but the clear view brought to us by the formal work, so the results will be expressed using the original mathematical objects of this logic.

Some problems of this logic are demonstrated, relatively to the representation of graph rewriting. Some are minor issues but some are far more important for the adequation between the formulas about graph rewriting and the actual rewriting systems. Invalidating some resulting propositions, solutions are given to reestablish the logical characterization of graph rewriting, which was the initial purpose.

Keywords. Dynamic Logic, Graph Rewriting, Adequation Issues

Key terms. Formal Method, Model

1 Introduction

Nearly every field of computer science uses graphs to represent data or the behavior of systems. Then, to get a higher level of dynamics, it uses rewriting in a more or less formal way to handle manipulation of such objects. In some fields, as in Formal Methods and Model-Driven Engineering, graphs are one of the main tools, having advanced methods to express and reason on these graphs is of great pertinance.

Modal logic allows to express relational properties naturally and, with Kripke semantics, is closely linked to graphs which are its models. Yet, instead of discussing graphs and rewriting using hyper-graphs as models, the transformed

[★] This work has been funded by the CLIMT project (ANR-11-BS02-016 of the French Agence Nationale de la Recherche)

graphs could be directly these models. Dynamic logic as defined by D. HAREL offers by its modalities the possibility to express relations on models when they have some desired properties. In this spirit, introduced during ICGT 2010, “A Dynamic Logic for Termgraph Rewriting” [1] was proposed by P. BALBIANI, R. ECHAHED and A. HERZIG as a suitable dynamic logic to describe graphs and transformations on these graphs.

The termgraph lifts the concept of term to the more general one of graph. It was initially introduced to represent terms while having a simpler way to talk about recursion or sharing of sub-terms, what a tree-like structure doesn’t allow to do easily.

Computer science involves data structures that are usually syntactically represented as simple terms. It is interesting to develop such a tool for more than just a graphic representation: using graph rewriting, research could be done on term rewriting with this rich and powerful layer of language of graph structure.

In the following, in *Section 2* we will present the type of graphs we use, then in *Section 3* the dynamic logic for graph rewriting, its syntax and its semantics. In *Section 4* the rewriting system is presented, and propositions to logically talk about it, but with issues to express these concepts with the logic as originally introduced. Graph homomorphisms, rewriting steps and application to a graph, with the definition of rewriting rules, matching of rules and normal forms, actually leads to some divergences between actual rewriting and its translation using the logic. *Section 4* also discuss and proposes some solutions for these original issues, for such an utilisation.

2 Termgraph And Rooted Termgraph

To illustrate terms representation as graphs from some typical λ -calculus, as shown in Figure 1, a classical tree-like term in (a), an example of argument sharing in (b), and eventually a representation for a fixpoint operator in (c), usually denoted Y for a function f [2]. One can easily unfold the recursion in the latter graph and get $Y f = f (Y f)$, which is what is expected from a fixpoint operator.

The graphs are deterministic and has labelled nodes and edges [3], but for convenience edge labels will be named features. A linear graph grammar is defined over a set of nodes \mathcal{N} , a set of labels Ω and a set of features \mathcal{F} [1], by the following rules:

$$\begin{aligned} \text{Node} &::= n : \omega (f_1 \Rightarrow \text{Node}, \dots, f_k \Rightarrow \text{Node}) \mid n : \bullet \mid n \\ &\text{avec } n \in \mathcal{N}, \omega \in \Omega \text{ et } f_1, \dots, f_k \in \mathcal{F} \\ \text{TermGraph} &::= \text{Node} \mid \text{Node} + \text{TermGraph} \end{aligned}$$

Allowing one to define a node, with its label and its direct sons, a node without label or just a reference to an actual node with the first rules, and to define multiparts termgraphs with the others.

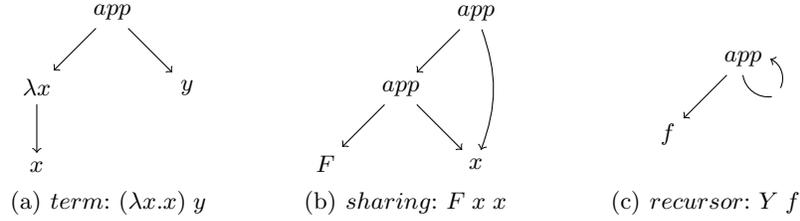


Fig. 1. Examples of term representation

In a less syntactic and maybe more semantic manner, a termgraph is defined as well by a structure.

$$(\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}) \text{ with}$$

- \mathcal{N} a finite set of nodes
- \mathcal{E} a finite set of edges
- $\mathcal{L}^{\mathcal{N}}$ a partial function from \mathcal{N} to Ω , associating a node to its label
- $\mathcal{L}^{\mathcal{E}}$ a total function from \mathcal{E} to \mathcal{F} , associating a edge to its feature
- \mathcal{S} a source function from \mathcal{E} to \mathcal{N}
- \mathcal{T} a target function from \mathcal{E} to \mathcal{N}

It's assumed that the previous definitions respect a determinism condition defined as

$$\forall e_1, e_2 \in \mathcal{E}, \mathcal{S}(e_1) = \mathcal{S}(e_2) \wedge \mathcal{L}^{\mathcal{E}}(e_1) = \mathcal{L}^{\mathcal{E}}(e_2) \rightarrow e_1 = e_2$$

Rooted TermGraphs. In the following, for the need of the logic, the graphs will be an extension of the termgraph with a specific node pointed as its root.

$$(\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r) \text{ with the root } r \in \mathcal{N}$$

3 Dynamic Logic for Graph Rewriting

A modal logic is a propositional logic, extended with one or several modality operators. Dynamic logic [4] is a multi-modal logic, its modalities being actions and the possibility to express a choice, an iteration or the sequence of actions. Actions can be defined to express graph transformations by miscellaneous modalities.

3.1 Syntax of the Dynamic Logic

For given countable sets \mathcal{F} and Ω , of features and labels (their respective elements being usually denoted a, b, \dots and ω, π, \dots), the rules defining the formulas and the actions of the syntax are the following. [1]

For an action α :

$\alpha ::= a$ for the navigation by $a \in \mathcal{F}$ from the root
 $| U$ for changing the root to any node in the graph
 $| n | \mathbf{n}$ for the creation of a node n , eventually setting it as the root
 $| \phi?$ for the verification of the validity of a formula ϕ for the root
 $| (\omega :=_l \phi) | (\omega :=_g \phi)$ for labeling a node with a label $\omega \in \Omega$ if a formula ϕ is valid, locally for the root or globally for any node.
 $| (a + (\phi, \psi)) | (a - (\phi, \psi))$ for adding or removing edges with the feature a , between nodes verifying a formula ϕ and those verifying a formula ψ .
 $| (\alpha_1; \alpha_2) | (\alpha_1 \cup \alpha_2) | \alpha_1^*$ for the definition of a sequence, a choice or an iteration over actions α_1 and α_2 .

For a formula ϕ :

$$\phi ::= \omega \mid \perp \mid \neg \phi \mid \phi \vee \psi \mid [\alpha] \phi$$

ω as a formula means that node labels can be atomic formulas. And intuitively, $[\alpha] \phi$ means that after any execution of an action α , the formula ϕ holds. The propositional logic of such dynamic logic being a classical one, some more symbols can be defined as usual conjunction, implication and equivalency being respectively $\phi \wedge \psi \equiv \neg(\neg \phi \vee \neg \psi)$, $\phi \rightarrow \psi \equiv \neg \phi \vee \psi$ and $\phi \leftrightarrow \psi \equiv (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$, for given formulas ϕ and ψ . One more modality can be defined, for a given action α and formula ϕ , as $\langle \alpha \rangle \phi \equiv \neg[\alpha] \neg \phi$. Intuitively, it holds when an execution of the action α makes ϕ hold.

3.2 Semantics of the Dynamic Logic

A semantic can be defined for this dynamic logic using rooted termgraphs as models and verifying the properties expressed by the formulas on them, using the sets of labels and features of the logic to define them. [1]

Models. The rooted termgraphs used here are different from the previous ones. The labeling function for nodes is now defined over the power set of Ω , so typed $\mathcal{L}^{\mathcal{N}} : \mathcal{N} \rightarrow \mathcal{P}(\Omega)$, of what the former definition is said to be a particular case.

Interpretation of Actions and Formulas. Before the definition of a satisfiability relation, it needs interpretation functions over the models.

I_G is defined as

- $I_G(a) = \{e \mid e \in \mathcal{E} \text{ and } \mathcal{L}^{\mathcal{E}}(e) = a\}$ the set of a edges.
- $I_G(\omega) = \{n \mid n \in \mathcal{N} \text{ and } \omega \in \mathcal{L}^{\mathcal{N}}(n)\}$ the set of nodes having the ω label.

And, for any $a \in \mathcal{F}$, R_G is a family of binary relations defined such as

- $R_G(a) = \{(n_1, n_2) \mid \exists e \in I_G(a), \mathcal{S}(e) = n_1 \text{ and } \mathcal{T}(e) = n_2\}$

Because of dependency between formulas and actions, the satisfiability relation \models for a formula F and a rooted termgraph G requires an inductive definition on F , dependent of a relation $G \longrightarrow_\alpha G'$ for every action α .

- $G \models \omega$ iff $n_0 \in I_G(\omega)$, interpreting ω in G .
- $G \not\models \perp$
- $G \models \neg\phi$ iff $G \not\models \phi$
- $G \models \phi \vee \psi$ iff $G \models \phi$ or $G \models \psi$
- $G \models [\alpha]\phi$ iff for any rooted termgraph G' , if $G \longrightarrow_\alpha G'$ then $G' \models \phi$

with $G \longrightarrow_\alpha G'$ the binary relation between two termgraphs G and G' considering the action α . It is defined inductively on α , with $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$ and $G' = (\mathcal{N}', \mathcal{E}', \mathcal{L}^{\mathcal{N}'}, \mathcal{L}^{\mathcal{E}'}, \mathcal{S}', \mathcal{T}', r')$, but only definitions for the useful cases will be introduced. The definitions are declarative, so it should be read as conditions for a correct relation between G and G' and not as the way to get a model G' from a model G .

A notation $\llbracket e \rrbracket_G$ is introduced for a graph G and an edge e of G , to express $(\mathcal{S}(e), \mathcal{L}^{\mathcal{E}}(e), \mathcal{T}(e))$. Ambiguity decreases, in the following definitions, the set \mathcal{E} being only identifiers of edges for the functions $\mathcal{L}^{\mathcal{E}}, \mathcal{S}$ and \mathcal{T} and not tuples of these informations. We can justify this by looking at the other way, with \mathcal{E} as a set of tuples, and seeing that it does not make much sense: for example, when redirecting edges, the set \mathcal{E} was staying the same, and so were the tuples in this set, but the target of an edge was changed and thus a tuple was associated by functions to information which is no longer the content of the tuples.

For convenience, another notation $G_{[n']}$ is introduced for a node n' of a graph $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$, to express the graph $(\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, n')$, or in more simple terms, to express changing the root of G with n' .

- $G \longrightarrow_a G'$ iff
- $\mathcal{N}' = \mathcal{N}, \mathcal{E}' = \mathcal{E}, \mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}, \mathcal{S}' = \mathcal{S}$ and $\mathcal{T}' = \mathcal{T}$, to express almost all parts of G' being the same.
- $(n_0, n'_0) \in R_G(a)$, to express the possibility to navigate from the root of G to the root of G' .

- $G \longrightarrow_U G'$ iff
- $\mathcal{N}' = \mathcal{N}, \mathcal{E}' = \mathcal{E}, \mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}, \mathcal{S}' = \mathcal{S}$ and $\mathcal{T}' = \mathcal{T}$, to express no characterization for the root of G' but other parts being the same.

- $G \longrightarrow_{(\omega:=g\phi)} G'$ iff
- $\mathcal{N}' = \mathcal{N}, \mathcal{E}' = \mathcal{E}, \mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}, \mathcal{S}' = \mathcal{S}, \mathcal{T}' = \mathcal{T}$ and $r' = r$
- for any $m \in \mathcal{N}$, if $G_{[m]} \models \phi$ then $\mathcal{L}^{\mathcal{N}'}(m) = \mathcal{L}^{\mathcal{N}}(m) \cup \{\omega\}$ else $\mathcal{L}^{\mathcal{N}'}(m) = \mathcal{L}^{\mathcal{N}}(m) \setminus \{\omega\}$, expressing the addition or deletion of the label ω of any node m of the graph satisfying the formula ϕ .

- $G \longrightarrow_{\phi?} G'$ iff

- $\mathcal{N}' = \mathcal{N}$, $\mathcal{E}' = \mathcal{E}$, $\mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}$, $\mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}$, $\mathcal{S}' = \mathcal{S}$, $\mathcal{T}' = \mathcal{T}$ and $r' = r$
- $G \models \phi$, to express the formula ϕ being valid for G .

$G \rightarrow_{(\omega:=l\phi)} G'$ iff

- $\mathcal{N}' = \mathcal{N}$, $\mathcal{E}' = \mathcal{E}$, $\mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}$, $\mathcal{S}' = \mathcal{S}$, $\mathcal{T}' = \mathcal{T}$ and $r' = r$
- if $G \models \phi$ then $\mathcal{L}^{\mathcal{N}'}(r) = \mathcal{L}^{\mathcal{N}}(r) \cup \{\omega\}$ else $\mathcal{L}^{\mathcal{N}'}(r) = \mathcal{L}^{\mathcal{N}}(r) \setminus \{\omega\}$, to express the addition or deletion of the label ω from the root.

$G \rightarrow_{(a+(\phi,\psi))} G'$ iff

- $\mathcal{N}' = \mathcal{N}$ and $\mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}$
- Considering the set of candidate edges $\mathcal{C} = \{(n_s, a, n_t) \mid \text{with } n_s \in \mathcal{N} \text{ and } n_t \in \mathcal{N} \text{ such as } G_{[n_s]} \models \phi \text{ and } G_{[n_t]} \models \psi\}$, to be added only between nodes validating the formulas ϕ and ψ .
- $\mathcal{E}' \supset \mathcal{E}$, and for all $e \in \mathcal{E}$, $\llbracket e \rrbracket_{G'} = \llbracket e \rrbracket_G$, characterizing an addition without any loss.
- for all $p \in \mathcal{C}$, $\exists e \in \mathcal{E}'$, $\llbracket e \rrbracket_{G'} = p$ and for all $e \in \mathcal{E}' \setminus \mathcal{E}$, $\llbracket e \rrbracket_{G'} \in \mathcal{C}$, expressing candidate edges being added in \mathcal{E}' but nothing else.

$G \rightarrow_{(a-(\phi,\psi))} G'$ iff

- $\mathcal{N}' = \mathcal{N}$ and $\mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}$
- Considering the set of deleted edges $E = \{e \mid e \in \mathcal{E} \text{ such as } \llbracket e \rrbracket_{G'} = (n_s, a, n_t) \text{ with } n_s \text{ and } n_t \text{ such as } G_{[n_s]} \models \phi \text{ and } G_{[n_t]} \models \psi\}$.
- $\mathcal{E}' = \mathcal{E} \setminus E$ characterizing deletion of edges only between nodes validating formulas ϕ and ψ .
- for all $e \in \mathcal{E}$, $\mathcal{L}^{\mathcal{E}'}(e) = \mathcal{L}^{\mathcal{E}}(e)$, $\mathcal{S}'(e) = \mathcal{S}(e)$, $\mathcal{T}'(e) = \mathcal{T}(e)$ and $r = r'$.

$G \rightarrow_{\alpha;\beta} G'$ iff

- there exists a rooted termgraph G'' such $G \rightarrow_{\alpha} G''$ and $G'' \rightarrow_{\beta} G'$.

$G \rightarrow_{\alpha^*} G'$ iff

- there is a rooted termgraph sequence $(G^{(0)}, \dots, G^{(k)})$ with $G^{(0)} = G$, $G^{(k)} = G'$ and for all $i \in \{0, \dots, k-1\}$, $G^{(i)} \rightarrow_{\alpha} G^{(i+1)}$.

Semantics of left over actions n , \mathbf{n} and $\alpha_1 \cup \alpha_2$ are in the original paper.

At this point, the logic allows to characterize classes of graphs using the satisfiability of formulas by these graphs as models. Everything goes pretty well, but issues come when dealing with rewriting and propositions made to talk about graph rewriting.

4 Actual Rewriting, and its Issues

The approach here is an algorithmic one: transformation actions are defined within a rewriting system and can be applied to a graph, alone or sequentially. It forms rules of rewriting that could be applied if an instantiation of the rule is found in a given graph. Such instance can be defined with a graph morphism which embeds the graph domain of the rule into the graph in which the rule could be applied.

4.1 Homomorphism of Graphs

A homomorphism of labelled graphs $h : G \rightarrow G'$ can be defined, given two rooted termgraphs $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$ and $G' = (\mathcal{N}', \mathcal{E}', \mathcal{L}^{\mathcal{N}'}, \mathcal{L}^{\mathcal{E}'}, \mathcal{S}', \mathcal{T}', r')$. Somewhat, only with a function $h^n : \mathcal{N} \rightarrow \mathcal{N}'$ preserving the labeling of nodes but equally preserving the source and target function for the edges, and thus the labelization of edges.

So $\forall m \in \mathcal{N}, \mathcal{L}^{\mathcal{N}'}(h^n(m)) = \mathcal{L}^{\mathcal{N}}(m)$ is mandatory and then because of the determinism condition satisfied by the graphs, there is no ambiguity on the conditions for the edges of the codomain graph G' . For any e of \mathcal{E} , it only requires one existing e' of \mathcal{E}' verifying $\mathcal{S}'(e') = h^n(\mathcal{S}(e))$, $\mathcal{L}^{\mathcal{E}'}(e') = \mathcal{L}^{\mathcal{E}}(e)$ and $\mathcal{T}'(e') = h^n(\mathcal{T}(e))$, and so with corresponding source and target while preserving the edge labelization, mandatory too for such homomorphism. There is no specific condition for correspondence of roots of the two termgraphs.

Examples of the original paper can be presented here, in Figure 2, to display some homomorphisms: morphisms h_2 and h_3 , between three graphs $B1$, $B2$ and $B3$ displaying the association of their nodes.

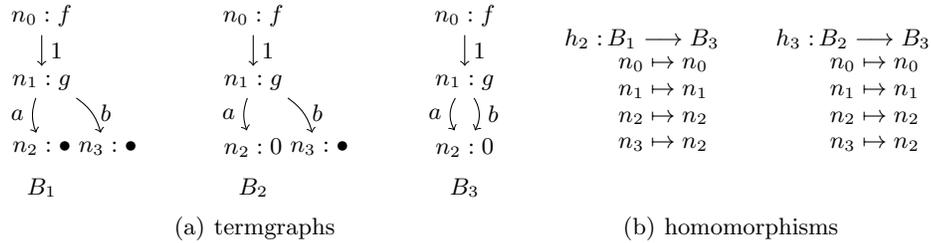


Fig. 2. Examples of termgraph homomorphisms

Existence of Graph Homomorphisms and Particularity of such Homomorphisms

In the original paper, a way to talk about homomorphisms on graphs using the logic is proposed, a formula express this concept and is defined for a graph $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$. Thus an action α_G and a formula ϕ_G can relate that there is a homomorphism from G to a graph G' if and only if $G' \models \langle \alpha_G \rangle \phi_G.[1]$

For this, considering $\mathcal{N} = \{n_0, \dots, n_{N-1}\}$, with N being the number of nodes of G and n_0 being its root, and considering a sequence $P = \{\pi_0, \dots, \pi_{N-1}\}$ of distinct elements of Ω , each π_i is going identify the node n_i .

The action α_G is defined

- with $\beta_G = (\pi_0 :=_g \perp) ; \dots ; (\pi_i :=_g \perp) ; \dots ; (\pi_{N-1} :=_g \perp)$, characterizing the elimination of any label π_i .

- with, for $0 \leq i \leq N - 1$, $\gamma_G^i = (\neg\pi_0 \wedge \dots \wedge \neg\pi_i)?; (\pi_i :=_l \top); U$, characterizing the labelization with π_i of a node not already labelled with π_k for $k \leq i$.
- and finally $\alpha_G = \beta_G; \gamma_G^0; \dots; \gamma_G^{N-1}$, sequencing these actions.

Then, the formula ϕ_G is defined

- with $0 \leq i \leq N - 1$, if $\mathcal{L}^{\mathcal{N}}(n_i) \neq \emptyset$ then $\psi_G^i = \langle U \rangle (\pi_i \wedge \mathcal{L}^{\mathcal{N}}(n_i))$ else $\psi_G^i = \top$, characterizing the conservation of the labelization of a node identified as the image of n_i , by the label π_i , but nothing if this node wasn't labelled.
- with for $0 \leq i, j \leq N - 1$, if $\exists e \in \mathcal{E}$ such $\mathcal{S}(e) = n_i$ and $\mathcal{T}(e) = n_j$ then $\zeta_G^{i,j} = \langle U \rangle (\pi_i \wedge \langle \mathcal{L}^{\mathcal{E}}(e) \rangle \pi_j)$ else $\zeta_G^{i,j} = \top$, characterizing the existence of edges corresponding to the ones of G .
- and finally $\phi_G = \psi_G^0 \wedge \dots \wedge \psi_G^{N-1} \wedge \zeta_G^{0,0} \wedge \dots \wedge \zeta_G^{N-1,N-1}$, verifying all these formulas.

In the latter definition, it is assumed to have a subset P of Ω of fresh labels not already used in G , and so that do not have to be preserved by the homomorphism in G' , and do not erase information when used in β_G . Such dedicated labels for identification of the elements of \mathcal{N} could be assumed as a part of the set Ω , by definition. This definition of P , identifying differently each node of G and the way it is used in the formula, requires more for a homomorphism than what implies the homomorphism definition. Acutally, the examples of the Figure 2, which do not stand against the definition, are not injective morphisms. In the formula $\langle \alpha_G \rangle \phi_G$, matched nodes are labelled to be identified to only one node of G , as the test $(\neg\pi_0 \wedge \dots \wedge \neg\pi_i)?$ express it. Therefore, it is mandatory for a homomorphism $h : G \longrightarrow G'$ to be injective to satisfy the formula and this necessary condition on the models. Although, such injectivity is a common feature of graph morphism in many graph rewriting frameworks, so it is not mandatory to define formally such a system using injective matching only and the initial definition of graph homomorphism should specify wether this particularity is actually required.

4.2 Rewriting Step and Translation in Logic

The rewriting is defined to be applied to a graph $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$ to obtain a graph $G' = (\mathcal{N}', \mathcal{E}', \mathcal{L}^{\mathcal{N}'}, \mathcal{L}^{\mathcal{E}'}, \mathcal{S}', \mathcal{T}', r')$, the result of the application of an action α to a graph G is denoted $\alpha[G]$. It is possible to make sequences of actions, as empty or the concatenation of an action α and another sequence, and the application of a sequence Δ to a graph G is denoted $\Delta[G].[1]$

- if Δ is the empty sequence then $\Delta[G] = G$
- if $\Delta = \alpha; \Delta'$ is a concatenation, with a sequential operator “;”, then $\Delta[G] = \Delta'[\alpha[G]]$

For a morphism h and a sequence Δ , $h(\Delta)$ denotes the sequence one obtains by substitution in Δ of any node n by $h(n)$.

The original paper proposes a way to talk about sequences of actions, by describing the sequence of rewriting actions as a sequence of logical actions. Thus, after the translation of an action α of the rewriting system, the relation $\longrightarrow_{tr(\alpha)}$ introduced with the semantic of the logic allows to relate models, the second potentially being the result of the application of the action to the first one. The translation of a sequence of a given action a and another sequence Δ will be the sequence of translations $\alpha_{a;\Delta} = \alpha_a; \alpha_\Delta$. Assuming that any action has a translation entirely independent of the rest of the sequence, the translation order actually does not matter, but a sequential translation seizes a correct idea.

For the rewriting actions:

$n \gg_a m$ is a local redirection.

An outgoing edge from a node n with the feature a is modified to point to a node m .

- $\mathcal{N}' = \mathcal{N}, \mathcal{E}' = \mathcal{E}, \mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}, \mathcal{S}' = \mathcal{S}$ and $r = r'$, the nodes, edges, their labelization, the source of the edges and the root are the same.
- for $e \in \mathcal{E}$ such $\mathcal{S}'(e) = n$ and $\mathcal{L}^{\mathcal{E}'}(e) = a$ then $\mathcal{T}'(e) = m$, the target of the only wanted edge is changed for m . $\forall e' \in \mathcal{E}'$, if $e' \neq e$ then $\mathcal{T}'(e') = \mathcal{T}(e')$, the target function doesn't change for the other edges.

To begin with, here is the given formula for the local redirection.

$$- \alpha_{n \gg_a m} = (a - (\pi_n, \top)); (a + (\pi_n, \pi_m))$$

One can notice that this formula depicts a transformation by deleting an edge, labelled a , between a node n and any other node, though the determinism dictates the existence of at most only one such edge. But above all thus an a edge between this node n and a node m is added at the end. Looking at the definition of the local redirection, one can see that $\mathcal{E}' = \mathcal{E}$ specify that no edge is actually added and the other item clearly specify that only an existing e of E with source as n and feature as a has its target changed to m .

A difference happens between the rewriting of a graph and the models linked by the actions of the translation of the rewriting action, as shown in Figure 3 which is a counter example for adequation between this translation and rewriting.

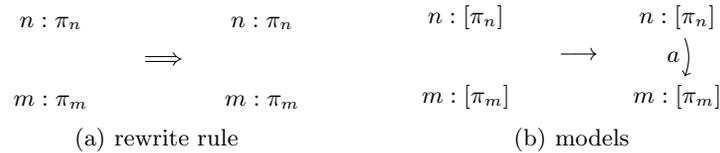


Fig. 3. Local Redirection Difference

The problem being that the logical actions don't require the existence of the redirected edge, what is obviously mandatory to rewrite it. We proposed, as a correction for this problem, the formula:

$$- \alpha_{n \gg_a m} = (\lambda_n :=_g \perp) ; (\lambda_n :=_g \pi_n \wedge \langle a \rangle \top) ; (a - (\pi_n, \top)) ; (a + (\lambda_n, \pi_m))$$

The added actions require the marking of a node n with a λ_n , assumed fresh for the graph, validating the possibility to navigate along a a edge to any node, then it proceeds to delete and add the edge only with this sole mark. If no edge was removed, no mark will be found on the node n , correcting this problem of adequation between rewriting and logical translation.

$n \gg m$ is a global redirection.

The target of every edge pointing to the node n is modified to make the edges point to the node m .

- $\mathcal{N}' = \mathcal{N}$, $\mathcal{L}^{\mathcal{E}'} = \mathcal{E}$, $\mathcal{L}^{\mathcal{N}'} = \mathcal{L}^{\mathcal{N}}$, $\mathcal{L}^{\mathcal{E}'} = \mathcal{L}^{\mathcal{E}}$ and $\mathcal{S}' = \mathcal{S}$, the nodes, edges, their labelizations and the source of the edges.
- for all $e \in \mathcal{E}$ such as $\mathcal{T}(e) = n$ then $\mathcal{T}'(e) = m$ else $\mathcal{T}'(e) = \mathcal{T}(e)$, the targets of the only wanted edges are changed for m , otherwise it does not change.
- if $n = r$ then $r' = m$ else $r' = r$, the root changes if it's the former target of the changed edges.

If one looks at the translation of the global redirection, with first a formula to globally redirect to a node m every a edges initially pointing to a node n :

$$- \alpha_{n \gg_a^g m} = (\lambda_a :=_g \perp) ; (\lambda_a :=_g \langle a \rangle \pi_n) ; (a - (\top, \pi_n)) ; (a + (\lambda_a, \pi_m))$$

This formula does not suffer of this problem, the second action marking with a λ_a any reachable node by an edge with the feature a . Then, the action adding the edges does not add wrongly an inexistant previously removed edge, since the label λ_a ensures these nodes are subjects to redirection. However, it is implicitly required that this λ_a is a dedicated label for this action, to not interfere with any other formula using this label for another purpose. The full translation of the rewriting step is finally a sequence of the previously defined formulas, for every feature of the graph:

$$- \alpha_{n \gg m} = ;_{a \in \mathcal{F}} \alpha_{n \gg_a^g m}$$

Comes finally the last rewriting action, the node labelization.

$n : \omega (f_1 \Rightarrow n_1, \dots, f_k \Rightarrow n_k)$ is a node definition or labelization.

It adds a node n , if it does not already belongs to the graph, or modifies an already existing one. It assigns the label ω and defines the edges e_1 to e_k outgoing of this node n , respectively pointing to the nodes n_1 to n_k with the labels f_1 to f_k , according to the following definition:

- $\mathcal{N}' = \mathcal{N} \cup \{n, n_1, \dots, n_k\}$, nodes of the rules which are not already included in G are added.

- $\mathcal{L}^{\mathcal{N}'}(n) = \omega$ and $\forall m \in \mathcal{N} \setminus \{n\}, \mathcal{L}^{\mathcal{N}'}(m) = \mathcal{L}^{\mathcal{N}}(m)$, n is labelled with ω and the other nodes keep the same labeling.
- Considering the newly defined edges $E = \{e_i \mid \mathcal{S}'(e_i) = n, \mathcal{L}^{\mathcal{E}'}(e_i) = f_i \text{ and } \mathcal{T}'(e_i) = n_i\}$ with $1 \leq i \leq k$.
- $\mathcal{E}' = \mathcal{E} \cup E$, new edges are added to the already existing ones.
- $\forall e_i \in E, \mathcal{L}^{\mathcal{E}'}(e_i) = f_i$, the features of the new edges are defined, and $\forall e \notin E, \mathcal{L}^{\mathcal{E}'}(e) = \mathcal{L}^{\mathcal{E}}(e)$, the features of the other edges don't change.
- $\forall e_i \in E, \mathcal{S}'(e_i) = n$ and $\forall e \notin E, \mathcal{S}'(e) = \mathcal{S}(e)$, the same thing is done for the source function.
- $\forall e_i \in E, \mathcal{T}'(e_i) = n_i$ and $\forall e \notin E, \mathcal{T}'(e) = \mathcal{T}(e)$, the same thing is done for the target function.
- $r' = r$, the root remains the same.

And the formula, initially given as its translation:

$$- \alpha_n : \omega (f_1 \Rightarrow n_1, \dots, f_k \Rightarrow n_k) = \\ U ; \pi_n ? ; (\omega :=_l \top) ; (f_1 + (\pi_n, \pi_{n_1})) ; \dots ; (f_k + (\pi_n, \pi_{n_k}))$$

One could there raise a first issue, regarding the end of the formula: it adds without much care outgoing edges of the node n , thereby not ensuring the determinism. The result is that no model can be related as resulting of the application of this action, when what was intended was a relation on models displaying a redirection of these edges. We proposed this correction, using as previously the idea of edge redirection by deletion of the former edge and addition of the redirected one.

$$- \alpha_n : \omega (f_1 \Rightarrow n_1, \dots, f_k \Rightarrow n_k) = \\ (f_1 - (\pi_n, \top)) ; \dots ; (f_k - (\pi_n, \top)) ; U ; \pi_n ? ; (\omega :=_l \top) ; \\ (f_1 + (\pi_n, \pi_{n_1})) ; \dots ; (f_k + (\pi_n, \pi_{n_k}))$$

Now, a model related by the relation of the corresponding formula for this action can exist, and it will be a deterministic graph with redirected edges. However, there still remains a problem. If one looks at the transformation expressed by the formula, it translates the labelization of a node positioning the root on a node n , identified by π_n , thus it labelizes it with ω and finally changes the edges outgoing of this node n . As displayed in Figure 4, this raises again a major difference between the graph one obtains by rewriting and the models related by the relation for the modality of the logical action. The behavior is not the same because the formula requires only the addition of a label to the multiple already existing labels of a node while the rewriting action requires the replacement as an unique label.

For a better understanding of the extent of this problem, one should look at the following, which demonstrates how much of an issue that becomes in regard of characterization of rewriting system.

$$\begin{array}{ccc}
 n : l & \Longrightarrow & n : \omega & n : [l] & \longrightarrow & n : [\omega, l] \\
 \text{(a) rewrite rule} & & & \text{(b) models} & &
 \end{array}$$

Fig. 4. Node Labeling Difference

4.3 Rewriting Rules and Rewriting System Characterization

Rewriting rules

A rewriting rule is expressed as a graph L and a sequence of actions Δ to apply, and will be noted (L, Δ) . It is said that a graph G is rewritten as a graph G' if there exists a homomorphism $h : L \rightarrow G$ such as $h(\Delta)[G] = G'$, denoted $G \rightarrow_{L, \Delta} G'$.

A simple yet useful example may be defined, representing an action on a simple on-off switch of an electrical network:

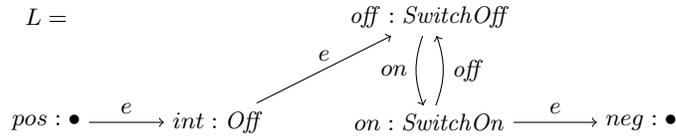


Fig. 5. Left-hand side L of a rule

Displayed in Figure 5, on the left of L there is a positive terminal of the circuit, on the right side there is a negative one. The edges labelled edges as e are for the electrical circuit and the switch is currently set to *Off*. The two positions *SwitchOn* and *SwitchOff* are linked to each other to avoid any mismatch if another switch is part of the network, even though a one-way link alone could be enough.

$$\Delta_{switchOn} = (int : On () ; int \gg_e on)$$

Fig. 6. Sequence Δ of a rule

The sequence for this rule, as displayed in Figure 6, is made of an action to mark the switch node with *On* without adding any extra edge, and an action to redirect the edge e , outgoing from this node to the *Off* position of the switch, to get a graph representing a closed electrical circuit, displayed now in Figure 7 as the result on the graph L .

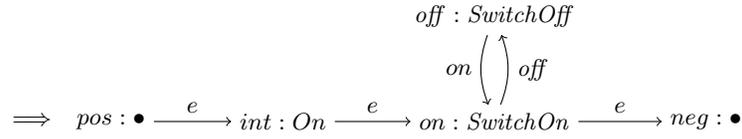


Fig. 7. result of applying Δ to L

Rewriting System Characterization and The Labelization Problem

Using the previous example of a switch activation, a graph G is displayed in the Figure 8, being a simple electrical network having only a generator and a switch, and being really similar to the left-hand side of the rule $(L, \Delta_{\text{switchOn}})$ previously defined.

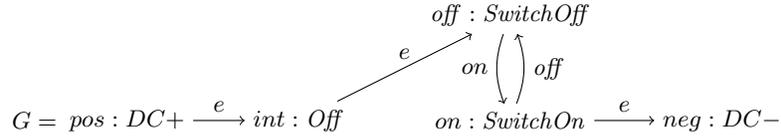


Fig. 8. graph G

The normal form of the graph G , with respect to the rule $(L, \Delta_{\text{switchOn}})$, satisfies the formula $\psi := [U] \neg \text{Off}$, because of its switch being activated and thus being labelled with On , so no node is labelled with Off anymore.

In the end of the original paper, the proposition 4 [1] which defines a way to talk about a normal formal of a graph G satisfying ψ says that this should be equivalent to the following satisfaction

$$G \models [(\alpha_L; \phi_L?; \alpha_{\Delta_{\text{switchOn}}})^*][(\alpha_L; \phi_L?) \perp \rightarrow \psi]$$

To explain this proposition, one can think about a normal form with respect of a rule as resulting of successive matchings and applications of this rule. When the rule cannot be matched anymore this implies that the formula ψ holds, which characterizes the normal forms. In the example with the rule $(L, \Delta_{\text{switchOn}})$, every switch must be activated and the rule shouldn't be matchable then, because there is no label Off or any e edge pointing to the right position. But because of the difference between rewriting and translation in the logic, when every e edge of the switch is redirected in the models, no more matching is available but there is still the label Off , along with the label On labeling the node int . The formula ψ is thus unsatisfied, the graph G and the formula ψ are a counter example for this proposition.

First, one can notice that relying on the definition of the models, the logic makes use of the labels to mark and reason, but without any difference between the ones being actual parts of the graph and the ones serving the logic. Even if the problem appears with the semantics, the solution is not there, because any change of the model definition, allowing to make a difference between graph and logic for example by simply splitting in two layer these informations, doesn't mean that the same syntax of the logic allows to express this difference.

The problem comes with the translation of the label definition of a node, it seems sounding to complete this translation to get the correspondence between rewriting and relation on models. But the rules and the matching use the labels to logically identify the node, so one should be careful with erasing, thus to erase everything is not an option here. Actually, the syntax of the logic needs explicitly the label to delete, with a global or local labeling action. And the syntax of the action currently does not allow to say which label will be erased. This is where the problem lies in: there is only the identifier of a node and the new label. To actually translate the action, it is possible if it is given more information, so a new label definition action is a possibility or the syntax of the logic should be changed to express actions on a new model type.

But before any syntactic change, of rewriting actions or logic, it is interesting to consider the action in the context of a graph. Because of the $\mathcal{L}^{\mathcal{N}} : \mathcal{N} \rightarrow \Omega$ of the graph, and having already the node n of the action, the labeling function can give the label to remove. Regarding of the action to translate, there is not this information, but more generally rewriting actions are defined for a specific graph. Thus, lifting the translation in formula to the level of the rule, this gives the required information. From this rule, a graph is given as left-hand side and thus a $\mathcal{L}^{\mathcal{N}} : \mathcal{N} \rightarrow \Omega$ function can be used, and it allows an explicit erasing of a label as it was implicitly expected by the rewriting action.

For a graph $G = (\mathcal{N}, \mathcal{E}, \mathcal{L}^{\mathcal{N}}, \mathcal{L}^{\mathcal{E}}, \mathcal{S}, \mathcal{T}, r)$ and a node labeling action $n : \omega (f_1 \Rightarrow n_1, \dots, f_k \Rightarrow n_k)$, we give as a correct translation the formula

$$\begin{aligned} - \alpha_{G, n : \omega (f_1 \Rightarrow n_1, \dots, f_k \Rightarrow n_k)} = \\ U ; \pi_n ? ; (\mathcal{L}^{\mathcal{N}}(n) :=_l \perp) ; (\omega :=_l \top) ; \\ (f_1 - (\pi_n, \top)) ; \dots ; (f_k - (\pi_n, \top)) ; (f_1 + (\pi_n, \pi_{n_1})) ; \dots ; (f_k + (\pi_n, \pi_{n_k})) \end{aligned}$$

However, the application of the rule to the left-hand side results in another graph with another labeling function and the translation of a sequence cannot remain as previously proposed. Every node labeling action should be translated depending on the result of previously applied actions. The translation of a sequence of an action a and another sequence Δ is a sequence of sequential translations:

$$\alpha_{G, (a; \Delta)} = \alpha_{G, a} ; \alpha_{a[G], \Delta}$$

One could notice that this translation is still done in a linear way through the sequence, but this definition could actually be slightly changed because every action but node labelization can be translated independently. Only the node labelization is dependent of a context and more precisely there are conflicts only

when editing the same node, having to erase a label which was just defined previously in the sequence.

5 Conclusion

This paper provided an introduction to a dynamic logic, defined by P. BALBIANI, R. ECHAHED and A. HERZIG. Originally, a formalization was done using the Coq proof assistant and so was done a study of this logic. During this work some mistakes were spotted. While sometimes these were just minor-looking imprecisions, when working with formal logic, it may be relevant to point out such ambiguities if the logic can become incoherent with a wrong interpretation.

Other mistakes are not only due to interpretation, and as it was demonstrated, they allow to find counter-examples for propositions and thus to establish incoherence with the main goal of this logic, to talk about rewriting systems. Solutions to these issues are proposed, they resolve the issues by getting back to an adequation between actual rewriting and relations on models of the logic.

A first idea to find another technical problem is the conservation of determinism of graphs, which only seems currently assumed and can be broken by the action of the logic, heading to the impossibility to relate a model to another because of the determinism, assumed by definition. It could be more explicit in the semantics of rewriting, and currently the definition of one of the rewriting actions allows to add edges breaking this condition. This is something the logic can express and handle during the translation of these rewriting actions, this looks like an interesting improvement of the logical framework.

The models are defined with a multi-labeling function of nodes and already demonstrate some differences during translation of rewriting into the logic, because the models are graphs with a unique-labeling function into the initial rewriting system. It is not totally clear whether the difference stops there, and it is required to study more uses of the logic in regards of rewriting to be sure of the correct use. It seems interesting to study the reverse translation as well, from logic to rewriting, to get an useful and complete logical framework, in the idea of *Curry – Howard* correspondence with terms as proofs.

References

1. Balbiani, P., Echahed, R., Herzig, A.: A dynamic logic for termgraph rewriting. In Ehrig, H., Rensink, A., Rozenberg, G., Schürr, A. (eds.) ICGT. LNCS, vol. 6372, pp.59–74. Springer, Heidelberg (2010)
2. Ariola, Z.M., Klop, J.W.: Lambda calculus with explicit recursion. *Inf. Comput.* 147(2), 154–233 (1997)
3. Barendregt, H., van Eekelen, M., Glauert, J., Kennaway, J., Plasmeijer, M., Sleep, M.: Term graph rewriting. In de Bakker, J., Nijman, A., Treleaven, P. (eds.) PARLE Parallel Architectures and Languages Europe. LNCS, vol. 259, pp. 141–158. Springer, Heidelberg (1987)
4. Harel, D., Kozen, D., Tiuryn, J.: Dynamic logic. *Handbook of Philosophical Logic*, MIT Press. 497–604 (1984)

Logical Foundations for Reasoning about Transformations of Knowledge Bases^{*} ^{**}

Mohamed Chaabani¹, Rachid Echahed² and Martin Strecker³

¹ LIMOSE, University of Boumerdès, Algeria

² Laboratoire d'Informatique de Grenoble

<http://membres-liglab.imag.fr/echahed/>

³ Université de Toulouse / IRIT

<http://www.irit.fr/~Martin.Strecker/>

Abstract. This paper is about transformations of knowledge bases with the aid of an imperative programming language which is non-standard in the sense that it features conditions (in loops and selection statements) that are description logic (DL) formulas, and a non-deterministic assignment statement (a choice operator given by a DL formula). We sketch an operational semantics of the proposed programming language and then develop a matching Hoare calculus whose pre- and post-conditions are again DL formulas. A major difficulty resides in showing that the formulas generated when calculating weakest preconditions remain within the chosen DL fragment. In particular, this concerns substitutions whose result is not directly representable. We therefore explicitly add substitution as a constructor of the logic and show how it can be eliminated by an interleaving with the rules of a traditional tableau calculus.

Keywords. Description Logic, Graph Transformation, Programming Language Semantics, Tableau Calculus

Key terms. MathematicalModel, SoftwareSystem, KnowledgeRepresentation

1 Introduction

Knowledge bases (KBs) are specific forms of graphs structures that are subject to change because the world they describe changes. The question explored by this paper is: What is an adequate formalism for describing these changes, and how to reason about the effects of changes?

Reasoning about graph transformations in full generality is hard [7]. Some decidable logics for graph transductions are known, such as MSO [6], but are

^{*} Preliminary workshop version of a paper to be presented at DL 2013

^{**} Part of this research has been supported by the *Climt* project (ANR-11-BS02-016).

descriptive, applicable to a limited number of graphs and often do not match with an algorithmic notion of transformation. Some implementations of verification environments for pointer manipulating programs exist [9], but they often impose severe restrictions on the kind of graphs that can be manipulated, such as having a clearly identified spanning tree.

In [4], the authors have introduced a dynamic logic which is very expressive. It has been designed to describe different kinds of elementary knowledge bases transformations (addition of new items, addition and deletion of links, etc.). It allows also to specify advanced properties on graph structures which go beyond mu-calculus or MSO logics. Unfortunately, the expressive power of that logic has a price: the undecidability of the logic. The purpose of the present paper is to identify a programming language together with a logic such that the transformation of the KB is decidable. The transformations themselves are not encoded in the logic itself (as in [4]) but in a dedicated imperative language for which we develop a Hoare-style calculus.

Work on (KB) updates [8] seem to approach the problem from the opposite direction: Add facts to a KB and transform the KB at the same time such that certain formulas remain satisfied. In our approach, the modification of the KB is exclusively specified by the program.

The work described in this paper is ongoing, some results are still preliminary. Based on previous work [5], we are in the process of coding the formalism described here in the Isabelle proof assistant [10]. Parts of the coding in this paper are inspired by formalizations in the Isabelle distribution and by [11]. The formal development accompanying this paper will be made available on the web⁴, which should also be consulted for proofs.

Before starting with the formal development, let us give an example of the kind of program (see Fig. 1) that we would like to write. Assume a knowledge base with objects of class A and B , and a relation r . The node n is initially connected to at least 3 objects of class A , and all objects it is connected to are of class B . Because the number of connections to A is too large, we execute a loop that selects an A -object (let's call it a) that n is connected to, and delete the r -connection between n and a . To compensate, we select an object b of class B and connect n to b . We stop as soon as the number of A -connections of n has reached 2, which is one of the post-conditions we can ascertain.

2 Logic

Our logic is a three-tier framework, the first level being DL *concepts* (“TBox”), the second level *facts* (“ABox”, instances of concepts), the third level *formulas* (Boolean combinations of facts and a simple form of quantification).

Concepts: We concentrate on a DL featuring concepts with simple roles and number restrictions, similar to \mathcal{ALCN} [2]. For c being the type of concept names

⁴ http://www.irit.fr/~Martin.Strecker/Publications/dl_transfo2013.html

```

vars n, a, b;

/* Pre: n : (≥ 3 r A) □ (∀ r B) */

while ( n : (> 2 r A) ) do {
  /* Inv: n : (≥ 2 r A) □ (∀ r B) */
  select a sth a : A ∧ (n r a);
  delete(n r a);
  select b sth b : B ;
  add(n r b)
}
/* Post: n : (= 2 r A) □ (∀ r B) */
    
```

Fig. 1. An example program

and r the type of role names, the data type C of concepts can be defined inductively by:

$$\begin{array}{l}
 C ::= c \quad (\text{atomic concept}) \\
 | \neg C \quad (\text{negation}) \\
 | C \sqcap C \quad (\text{conjunction}) \\
 | C \sqcup C \quad (\text{disjunction}) \\
 | (\geq n r C) \quad (\text{at least}) \\
 | (< n r C) \quad (\text{no more than}) \\
 | C[r := RE] \quad (\text{explicit substitution})
 \end{array}$$

Adding a universal concept \top and an empty concept \perp would not add expressivity, as they are equivalent to $(\geq 0 r C)$ respectively $(< 0 r C)$ for arbitrary r and C , and we will use them as shortcuts. We also write $(\exists r C)$ for $(\geq 1 r C)$ and $(\forall r C)$ for $(< 1 r (\neg C))$.

The last constructor, *explicit substitution* [1], is a particularity of our framework, required for a lazy elimination of substitutions that replace, in a concept C , a role name r by a role expression RE . If i is the set of individual variable names, the type RE is defined by

$$\begin{array}{l}
 RE ::= r \quad (\text{atomic role}) \\
 | r - (i, i) \quad (\text{deletion of relation instance}) \\
 | r + (i, i) \quad (\text{insertion of relation instance})
 \end{array}$$

Please note that concepts implicitly depend on the types c , r and i , which we assume mutually disjoint. A substitution can therefore never affect an individual variable.

A set-theoretic semantics is provided by a domain Δ an interpretation function \mathcal{I} mapping c to a set of individuals (subsets of Δ), r to a binary relation of individuals (subsets of $\Delta \times \Delta$), and i to individual elements of Δ .

For interpretation of concepts C , negation is inductively interpreted as complement, concept conjunction as intersection and disjunction as union. $\mathcal{I}(\geq$

$n \ r \ C) = \{x \mid \text{card}\{y \mid (x, y) \in \mathcal{I}(r) \wedge y \in \mathcal{I}(C)\} \geq n\}$, and analogously for $\mathcal{I}(< n \ r \ C)$. Here, *card* is the cardinality of finite sets (and 0 otherwise).

For interpretation of role expressions RE , we define $\mathcal{I}(r - (i_1, i_2)) = \mathcal{I}(r) - \{(\mathcal{I}(i_1), \mathcal{I}(i_2))\}$, and $\mathcal{I}(r + (i_1, i_2)) = \mathcal{I}(r) \cup \{(\mathcal{I}(i_1), \mathcal{I}(i_2))\}$.

Interpretation update $\mathcal{I}^{[r:=rl]}$ modifies the interpretation \mathcal{I} at relation name r to relation rl , thus $\mathcal{I}^{[r:=rl]}(r) = rl$ and $\mathcal{I}^{[r:=rl]}(r') = \mathcal{I}(r')$ for $r' \neq r$. With this, we can define the semantics of explicit substitution by $\mathcal{I}(C[r := RE]) = \mathcal{I}^{[r:=\mathcal{I}(RE)]}(C)$.

Facts: Facts make assertions about an instance being an element of a concept, and about being in a relation. In DL parlance, facts are elements of an ABox. The type of facts is defined as follows:

$$\begin{aligned} \text{fact} ::= & i : C && (\text{instance of concept}) \\ & | \ i \ r \ i && (\text{instance of role}) \\ & | \ i \ (\neg r) \ i && (\text{instance of role complement}) \\ & | \ i = i && (\text{equality of instances}) \\ & | \ i \neq i && (\text{inequality of instances}) \end{aligned}$$

The interpretation of a fact is a truth value, defined by:

$$\begin{aligned} - \mathcal{I}(i : C) &= (\mathcal{I}(i) \in \mathcal{I}(C)) \\ - \mathcal{I}(i_1 \ r \ i_2) &= (\mathcal{I}(i_1), \mathcal{I}(i_2)) \in \mathcal{I}(r) \text{ and } \mathcal{I}(i_1 \ (\neg r) \ i_2) = (\mathcal{I}(i_1), \mathcal{I}(i_2)) \notin \mathcal{I}(r) \\ - \mathcal{I}(i_1 = i_2) &= (\mathcal{I}(i_1) = \mathcal{I}(i_2)) \text{ and } \mathcal{I}(i_1 \neq i_2) = (\mathcal{I}(i_1) \neq \mathcal{I}(i_2)) \end{aligned}$$

Please note that since concepts are closed by complement, facts are closed by negation (the negation of a fact is again representable as a fact), and this is the main motivation for introducing the constructors “instance of role complement” and “inequality of instances”.

Formulas: A formula is a Boolean combination of facts. We also allow quantification over individuals i (but not over relations or concepts), and, again, have a constructor for explicit substitution.

$$\begin{aligned} \text{form} ::= & \perp \\ & | \ \text{fact} \\ & | \ \neg \text{form} \\ & | \ \text{form} \wedge \text{form} \quad | \ \text{form} \vee \text{form} \\ & | \ \forall i. \text{form} \quad | \ \exists i. \text{form} \\ & | \ \text{form}[r := RE] \end{aligned}$$

The extension of interpretations from facts to formulas is standard; the interpretation of substitution in formulas is in entire analogy to concepts. As usual, a formula that is true under all interpretations is called *valid*.

When calculating weakest preconditions (in Sect. 4), we obtain formulas which essentially contain no existential quantifiers; we keep them as constructor because they can occur as intermediate result of computations. We say that a formula is *essentially universally quantified* if \forall only occurs below an even and \exists only below an odd number of negations. For example, $\neg(\exists x. x : C \wedge \neg(\forall y. y : D))$ is essentially universally quantified.

Implication $f_1 \longrightarrow f_2$ is the abbreviation for $\neg f_1 \vee f_2$, and $ite(c, t, e)$ the abbreviation for $(c \longrightarrow t) \wedge (\neg c \longrightarrow e)$, not to be confused with the if-then-else statement presented in Sect. 3.

3 Programming Language

The programming language is an imperative language manipulating relational structures. Its distinctive features are conditions (in conditional statements and loops) that are restricted DL formulas, in the sense of Sect. 2. It has a non-deterministic assignment statement allowing to select an element satisfying a fact. Traditional types (numbers, inductive types) are not provided.

In this paper, we only consider a core language with traditional control flow constructs, but without procedures. Also, it is only possible to modify a relational structure, but not to “create objects” (with a sort of **new** statement) or to “deallocate” them. These constructs are left for further investigation.

3.1 Syntax

The type of statements is defined by:

$stmt ::= \text{Skip}$	(empty statement)
$\text{select } i \text{ sth } form$	(assignment)
$\text{delrel}(i \ r \ i)$	(delete arc in relation)
$\text{insrel}(i \ r \ i)$	(insert arc in relation)
$stmt \ ; \ stmt$	(sequence)
$\text{if } form \ \text{then } stmt \ \text{else } stmt$	
$\text{while } form \ \text{do } stmt$	

3.2 Semantics

The semantics is a big-step semantics with rules of the form $(st, \sigma) \Rightarrow \sigma'$ expressing that executing statement st in state σ produces a new state σ' .

The rules of the semantics are given in the Fig. 2. Beware that we overload logical symbols such as \exists , \wedge and \neg for use in the meta-syntax and as constructors of $form$.

The state space σ is in fact identical to an interpretation function \mathcal{I} as introduced in Sect. 2, and it is only in keeping with traditional notation in semantics that we use the symbol σ . We may therefore write $\sigma(b)$ to evaluate the condition b (a formula) in state σ .

Most of the rules are standard, apart from the fact that we do not use expressions, but formulas as conditions. The auxiliary function $delete_edge$ modifies the state σ by removing an r -edge between the elements represented by v_1 and v_2 . With the update function for interpretations introduced in Sect. 2, one defines

$$delete_edge \ v_1 \ r \ v_2 \ \sigma = \sigma^{[r := \sigma(r) - \{(\sigma(v_1), \sigma(v_2))\}]}$$

$$\begin{array}{c}
\frac{}{(\mathbf{Skip}, \sigma) \Rightarrow \sigma} \text{ (Skip)} \quad \frac{(c_1, \sigma) \Rightarrow \sigma'' \quad (c_2, \sigma'') \Rightarrow \sigma'}{(c_1; c_2, \sigma) \Rightarrow \sigma'} \text{ (Seq)} \\
\\
\frac{\sigma' = \text{delete_edge } v_1 r v_2 \sigma}{(\text{delrel}(v_1 r v_2), \sigma) \Rightarrow \sigma'} \text{ (EDel)} \quad \frac{\sigma' = \text{generate_edge } v_1 r v_2 \sigma}{(\text{insrel}(v_1 r v_2), \sigma) \Rightarrow \sigma'} \text{ (EGen)} \\
\\
\frac{\exists vi. (\sigma' = \sigma^{[v:=vi]} \wedge \sigma'(b))}{(\mathbf{select } v \text{ sth } b, \sigma) \Rightarrow \sigma'} \text{ (SelAssT)} \\
\\
\frac{\sigma(b) \quad (c_1, \sigma) \Rightarrow \sigma'}{(\mathbf{if } b \text{ then } c_1 \text{ else } c_2, \sigma) \Rightarrow \sigma'} \text{ (IfT)} \quad \frac{\neg \sigma(b) \quad (c_2, \sigma) \Rightarrow \sigma'}{(\mathbf{if } b \text{ then } c_1 \text{ else } c_2, \sigma) \Rightarrow \sigma'} \text{ (IfF)} \\
\\
\frac{\sigma(b) \quad (c, \sigma) \Rightarrow \sigma'' \quad (\mathbf{while } b \text{ do } c, \sigma'') \Rightarrow \sigma'}{(\mathbf{while } b \text{ do } c, \sigma) \Rightarrow \sigma'} \text{ (WT)} \quad \frac{\neg \sigma(b)}{(\mathbf{while } b \text{ do } c, \sigma) \Rightarrow \sigma} \text{ (WF)}
\end{array}$$

Fig. 2. Big-step semantics rules

and similarly

$$\text{generate_edge } v_1 r v_2 \sigma = \sigma^{[r:=\sigma(r) \cup \{(\sigma(v_1), \sigma(v_2))\}]}$$

The statement **select** v **sth** $F(v)$ selects an element vi that satisfies formula F , and assigns it to v . For example, **select** a **sth** $a : A \wedge (a r b)$ selects an element a instance of concept A and being r -related with a given element b .

select is a generalization of a traditional assignment statement. There may be several instances that satisfy F , and the expressiveness of the logic might not suffice to distinguish them. In this case, any such element is selected, non-deterministically. Let us spell out the precondition of (*SelAssT*): Here, $\sigma^{[v:=vi]}$ is an interpretation update for individuals, modifying σ at individual name $v \in i$ with an instance $vi \in \Delta$, similar to the interpretation update for relations seen before. We therefore pick an instance vi , check whether the formula b would be satisfied under this choice, and if it is the case, keep this assignment.

In case no satisfying instance exists, the semantics blocks, *i.e.* the given state does not have a successor state, which can be considered as an error situation. Some alternatives to this design choice can be envisaged: We might treat a **select** v **sth** $F(v)$ with unsatisfiable F as equivalent to a **Skip**. This would give us a choice of two rules, one in which the precondition of rule (*SelAssT*) is satisfied, and one in which it is not. As will be seen in Sect. 4, this would introduce essentially existentially quantified variables in our formulas when computing

weakest preconditions and lead us out of the fragment that we can deal with in our decision procedure. Alternatively, we could apply an extended type check verifying that select-predicates are always satisfiable, and thus ensure that type-correct programs do not block. This is the alternative we prefer; details still have to be worked out.

4 Weakest Preconditions

We compute weakest preconditions wp and verification conditions vc . Both take a statement and a DL formula as argument and produce a DL formula. For this purpose, while loops have to be annotated with loop invariants, and the `while` constructor becomes: `while {form} form do stmt`. Here, the first formula (in braces) is the invariant, the second formula the termination condition. The two functions are defined by primitive recursion over statements, see Fig. 3.

$ \begin{aligned} wp(\text{Skip}, Q) &= Q \\ wp(\text{delrel}(v_1 \ r \ v_2), Q) &= Q[r := r - (v_1, v_2)] \\ wp(\text{insrel}(v_1 \ r \ v_2), Q) &= Q[r := r + (v_1, v_2)] \\ wp(\text{select } v \ \text{sth } b, Q) &= \forall v. (b \longrightarrow Q) \\ wp(c_1; c_2, Q) &= wp(c_1, wp(c_2, Q)) \\ wp(\text{if } b \ \text{then } c_1 \ \text{else } c_2, Q) &= \text{ite}(b, wp(c_1, Q), wp(c_2, Q)) \\ wp(\text{while}\{iv\} \ b \ \text{do } c, Q) &= iv \\ \\ vc(\text{Skip}, Q) &= \top \\ vc(\text{delrel}(v_1 \ r \ v_2), Q) &= \top \\ vc(\text{insrel}(v_1 \ r \ v_2), Q) &= \top \\ vc(\text{select } v \ \text{sth } b, Q) &= \top \\ vc(c_1; c_2, Q) &= vc(c_1, wp(c_2, Q)) \wedge vc(c_2, Q) \\ vc(\text{if } b \ \text{then } c_1 \ \text{else } c_2, Q) &= vc(c_1, Q) \wedge vc(c_2, Q) \\ vc(\text{while}\{iv\} \ b \ \text{do } c, Q) &= (iv \wedge \neg b \longrightarrow Q) \wedge (iv \wedge b \longrightarrow wp(c, iv)) \wedge vc(c, iv) \end{aligned} $
--

Fig. 3. Weakest preconditions and verification conditions

Without going further into program semantics issues, let us only state the following soundness result that relates the operational semantics and the functions wp and vc :

Theorem 1 (Soundness). *If $vc(c, Q)$ is valid and $(c, \sigma) \Rightarrow \sigma'$, then $\sigma(wp(c, Q))$ implies $\sigma'(Q)$.*

What is more relevant for our purposes is the structure of the formulas generated by wp and vc , because it has an impact on the decision procedure. Besides the notion of essentially universally quantified introduced in Sect. 2, we need

the notion of *quantifier-free* formula: A formula not containing a quantifier. In extension, we say that a statement is quantifier-free if all of its formulas are quantifier-free.

By induction on c , one shows:

Lemma 1 (Universally quantified). *Let Q be essentially universally quantified and c be a quantifier-free statement. Then $wp(c, Q)$ and $vc(c, Q)$ are essentially universally quantified.*

5 Decision Procedure

5.1 Overview

We present a decision procedure for verifying the validity of essentially universally quantified formulas. As seen in Lemma 1, this is the format of formulas extracted by wp and vc , and as motivated by the soundness result (Theorem 1), validity of verification conditions is a precondition for ensuring that a program executes according to its specification.

Given an essentially universally quantified formula e , the rough lines of the procedure for determining that e is valid are spelled out in the following.

Getting rid of quantifiers:

1. Convert e to an equivalent prenex normal form p , which will consist of a prefix of universal quantifiers, and a quantifier-free body: $\forall x_1 \dots x_n. b$
2. p is valid iff its universal closure $ucl(p)$ (universal abstraction over all free variables of p) is.
3. Show the validity of $ucl(p)$ by showing the unsatisfiability of $\neg ucl(p)$.
4. $\neg ucl(p)$ has the form $\neg \forall v_1 \dots v_k, x_1 \dots x_n. b$. Pull negation inside the universal quantifier prefix, remove the resulting existential quantifier prefix, and show unsatisfiability of $\neg b$ with the aid of an extended tableau method.

Computation of prenex normal forms is standard. Care has to be taken to avoid capture of free variables, by renaming bound variables. Free variables are defined as usual; the free variables of a substitution $f[r := r - (v_1, v_2)]$ are those of f and in addition v_1 and v_2 (similarly for edge insertion). We illustrate the problem with the following program fragment prg :

```
select a sth a : A ;
select b sth b r a ;
select a sth a r b
```

For a given post-condition Q , we obtain

$$wp(prg, Q) = \forall a. a : A \longrightarrow \forall b. (b r a) \longrightarrow \forall a. (a r b) \longrightarrow Q$$

whose prenex normal form $\forall a_1, b, a_2. (a_1 : A \longrightarrow (b r a_1) \longrightarrow (a_2 r b) \longrightarrow Q)$ contains more logical variables than prg contains program variables.

Extended tableau method – prerequisites: The tableau method takes a quantifier-free formula f and proves its unsatisfiability or displays a model. We aim at reusing existing tableau methods (such as [3]) as much as possible. The difficulty consists in getting rid of the substitution constructor.

Substitution is compatible with the constructors of formulas:

Lemma 2 (Substitution in formulas).

$$\begin{aligned} \perp[r := re] &= \perp \\ (\neg f)[r := re] &= (\neg f[r := re]) \\ (f_1 \wedge f_2)[r := re] &= (f_1[r := re] \wedge f_2[r := re]) \\ (f_1 \vee f_2)[r := re] &= (f_1[r := re] \vee f_2[r := re]) \end{aligned}$$

The case of formulas which are facts, missing in Lemma 2, will be dealt with separately. This is due to the fact that substitution is not compatible with concepts, as will be seen in Sect. 5.2: For a given concept C , there is not necessarily a concept $C' = C[r := re]$. However, substitutions can be eliminated from facts, by the equations given in Sect. 5.2.

We will refer to the equations in Lemma 2 and those in Sect. 5.2 as *substitution elimination rules*. We say that a substitution in a formula is *visible* if one of these rules is applicable; and that it is *hidden* if none of these rules is applicable. For example, the substitution in $(x : (C_1 \sqcap C_2))[r := re]$ is visible; it is hidden in $(x : (C_1[r := re] \sqcap C_2[r := re]))$ and only becomes visible after application of an appropriate tableau rule, for example of the system \mathcal{ALCN} .

To describe our procedure, we introduce the following terminology: An ABox is a finite set of facts (interpreted as the conjunction of its facts), and a tableau a finite set of ABoxes (interpreted as a disjunction of its ABoxes). We need the following functions:

- *push_subst* takes a formula and applies substitution elimination rules as far as possible;
- *form_to_tab* converts to disjunctive normal form and then performs the obvious translation to a tableau;
- *tab_to_form* takes a tableau and constructs the corresponding formula.

Extended tableau method – procedure: Our method is parameterized by the following interface of an implementation of your favorite tableau calculus:

- a transition system $\mathcal{T} \Longrightarrow \mathcal{T}'$, defining a one-step transformation of a tableau \mathcal{T} to a tableau \mathcal{T}' .
- a function *sat* which checks, for tableaux \mathcal{T} that are irreducible wrt. \Longrightarrow , whether \mathcal{T} is satisfiable.

From this, we construct a restricted relation $\mathcal{T} \Longrightarrow_r \mathcal{T}'$, which is the same as \Longrightarrow provided that \mathcal{T} does not contain visible substitutions:

$$\frac{\mathcal{T} \Longrightarrow \mathcal{T}' \quad \text{no visible subst in } \mathcal{T}}{\mathcal{T} \Longrightarrow_r \mathcal{T}'}$$

We also define a relation \Longrightarrow^s that pushes substitutions until they become hidden:

$$\frac{\mathcal{T} \text{ contains visible subst} \quad \mathcal{T}' = \text{form_to_tab}(\text{push_subst}(\text{tab_to_form}(\mathcal{T})))}{\mathcal{T} \Longrightarrow^s \mathcal{T}'}$$

From these, we define the relation $\Longrightarrow_r^s = (\Longrightarrow_r \cup \Longrightarrow^s)$.

The extended tableau algorithm takes a formula f and computes a \mathcal{T}_f such that $\text{form_to_tab}(f)(\Longrightarrow_r^s)^* \mathcal{T}_f$. The result of the algorithm is $\text{sat}(\mathcal{T}_f)$.

The following lemmas show that \Longrightarrow_r^s is a correct and complete algorithm for deciding the decidability of formulas with substitution provided \Longrightarrow is for substitution-free formulas.

Lemma 3 (Termination). \Longrightarrow_r^s is well-founded provided \Longrightarrow is.

To show termination of the extended algorithm, define

- the *substitution size* of a *formula* or *fact* as the sum of the term sizes below its substitutions.
- the substitution size of a *tableau* as the multiset of the substitution sizes of its facts.

Note that application of \Longrightarrow^s leads to a reduction of the substitution size. For a well-founded measure m of \Longrightarrow , construct a well-founded measure of \Longrightarrow_r^s as the lexicographic order of the substitution size and m .

Lemma 4 (Confluence). \Longrightarrow_r^s is confluent provided \Longrightarrow is.

\Longrightarrow_r^s has no other critical pairs than \Longrightarrow .

Lemma 5 (Satisfiability). \Longrightarrow_r^s preserves satisfiability provided \Longrightarrow does.

The three auxiliary functions used for defining \Longrightarrow^s do.

5.2 Elimination of Substitutions

We now show how substitutions can be pushed into facts.

The constructors equality and inequality are easiest to handle:

- $(x = y)[r := re]$ reduces to $(x = y)$
- $(x \neq y)[r := re]$ reduces to $(x \neq y)$

For positive resp. negative instances of roles, we have:

- $(x r y)[r := r - (v_1, v_2)]$ reduces to $(\neg((x = v_1) \wedge (y = v_2))) \wedge (x r y)$
- $(x (\neg r) y)[r := r - (v_1, v_2)]$ reduces to $((x = v_1) \wedge (y = v_2)) \vee (x (\neg r) y)$
- $(x r y)[r := r + (v_1, v_2)]$ reduces to $((x = v_1) \wedge (y = v_2)) \vee (x r y)$
- $(x (\neg r) y)[r := r + (v_1, v_2)]$ reduces to $(\neg((x = v_1) \wedge (y = v_2))) \wedge (x (\neg r) y)$

whereas substitutions $(x \ r \ y)[r' := re]$ and $(x \ (\neg r) \ y)[r' := re]$ for $r \neq r'$ are the identity.

For facts of the form $x : C$, where C is a concept, we have the cases:

- $(x : \neg C)[r := re]$ reduces to $x : (\neg C[r := re])$
- $(x : C_1 \wedge C_2)[r := re]$ reduces to $x : (C_1[r := re] \wedge C_2[r := re])$
- $(x : C_1 \vee C_2)[r := re]$ reduces to $x : (C_1[r := re] \vee C_2[r := re])$
- $(x : (\geq n \ r \ C))[r' := re]$, for $r' \neq r$, reduces to $x : (\geq n \ r \ C[r' := re])$, and similarly when replacing \geq by $<$
- $(x : (\geq n \ r \ C))[r := r - (v_1, v_2)]$ reduces to

$$\begin{aligned} &ite((x = v_1) \wedge (v_2 : (C[r := r - (v_1, v_2)])) \wedge (v_1 \ r \ v_2), \\ &\quad (x : (\geq (n+1) \ r \ (C[r := r - (v_1, v_2)]))), \\ &\quad (x : (\geq n \ r \ (C[r := r - (v_1, v_2)])))) \end{aligned}$$

and similarly when replacing \geq by $<$

- $(x : (\geq (n+1) \ r \ C))[r := r + (v_1, v_2)]$ reduces to

$$\begin{aligned} &ite((x = v_1) \wedge (v_2 : (C[r := r + (v_1, v_2)])) \wedge (v_1 \ (\neg r) \ v_2), \\ &\quad (x : (\geq n \ r \ (C[r := r + (v_1, v_2)]))), \\ &\quad (x : (\geq (n+1) \ r \ (C[r := r + (v_1, v_2)])))) \end{aligned}$$

and similarly when replacing \geq by $<$

- $(x : (\geq 0 \ r \ C))[r := r + (v_1, v_2)]$ reduces to \top
- $(x : (< 0 \ r \ C))[r := r + (v_1, v_2)]$ reduces to \perp
- Pathological case $(x : C[subst_1])[subst_2]$: lift inner substitution to $(x : C)[subst_1][subst_2]$, then apply the above.

6 Conclusions

This paper proposes a language for rewriting knowledge bases, and methods for reasoning about the correctness of these programs, by means of a Hoare-style calculus. DL formulas are directly integrated into the statements of the programming language. The verification conditions extracted from these programs has been shown to be decidable, by a modular extension of existing tableau algorithms.

The work described here is still preliminary, in several respects, and the following points indicate directions for future investigations:

- We are in the process of coding the theory in the Isabelle proof assistant. Some parts of the proofs of Sect. 4 and most of Sect. 5.1 still has to be done. The purpose is to obtain a framework that will allow us to experiment more easily with variations of the logic.
- We have currently focused on the logic \mathcal{ALCN} . It is interesting to consider both less expressive logics (which offer more space for optimizations) and more expressive logics (to explore decidability questions). The process described in Sect. 5.1 is rather generic, but it remains to be seen whether more expressive DLs, featuring more complex role expressions, can be accommodated.

- In any case, the proof procedure sketched in Sect. 5 is rather of a theoretical than a practical value; an efficient implementation should not convert between formulas and tableaux as indiscriminately as suggested there, but apply propagation of substitutions locally.
- In a similar vein, it would be interesting to implement a transformation engine on the basis of the language described here, also with the purpose of evaluating the practical expressiveness of the language on larger examples.

References

1. Abadi, M., Cardelli, L., Curien, P.L., Lévy, J.J.: Explicit substitutions. *Journal of Functional Programming* 1(4), 375–416 (October 1991)
2. Baader, F., Sattler, U.: Expressive number restrictions in description logics. *Journal of Logic and Computation* 9(3), 319–350 (1999)
3. Baader, F., Sattler, U.: Tableau algorithms for description logics. In: Dyckhoff, R. (ed.) *Automated Reasoning with Analytic Tableaux and Related Methods*, Lecture Notes in Computer Science, vol. 1847, pp. 1–18. Springer Berlin / Heidelberg (2000)
4. Balbiani, P., Echahed, R., Herzig, A.: A dynamic logic for termgraph rewriting. In: 5th International Conference on Graph Transformations (ICGT). *Lecture Notes in Computer Science*, vol. 6372, pp. 59–74. Springer (2010)
5. Chaabani, M., Mezghiche, M., Strecker, M.: Vérification d’une méthode de preuve pour la logique de description \mathcal{ALC} . In: Ait-Ameur, Y. (ed.) *Proc. 10ème Journées Approches Formelles dans l’Assistance au Développement de Logiciels (AFADL)*. pp. 149–163 (Jun 2010)
6. Courcelle, B., Engelfriet, J.: *Graph structure and monadic second-order logic, a language theoretic approach*. Cambridge University Press (2011)
7. Immerman, N., Rabinovich, A., Reps, T., Sagiv, M., Yorsh, G.: The boundary between decidability and undecidability for transitive-closure logics. In: Marcinkowski, J., Tarlecki, A. (eds.) *Computer Science Logic*, Lecture Notes in Computer Science, vol. 3210, pp. 160–174. Springer Berlin / Heidelberg (2004)
8. Liu, H., Lutz, C., Milicic, M., Wolter, F.: Foundations of instance level updates in expressive description logics. *Artificial Intelligence* 175(18), 2170–2197 (2011)
9. Møller, A., Schwartzbach, M.I.: The pointer assertion logic engine. In: *PLDI*. pp. 221–231 (2001)
10. Nipkow, T., Paulson, L., Wenzel, M.: *Isabelle/HOL. A Proof Assistant for Higher-Order Logic*, Lecture Notes in Computer Science, vol. 2283. Springer Berlin / Heidelberg (2002)
11. Schirmer, N.: *Verification of Sequential Imperative Programs in Isabelle/HOL*. Ph.D. thesis, Technische Universität München (2006)

Program Algebras with Monotone Floyd-Hoare Composition

Andrii Kryvolap¹, Mykola Nikitchenko¹ and Wolfgang Schreiner²

¹ Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

kryvolapa@gmail.com, nikitchenko@unicyb.kiev.ua

² Johannes Kepler University, Linz, Austria

Wolfgang.Schreiner@risc.jku.at

Abstract. In the paper special program algebras of partial predicates and functions are described. Such algebras form a semantic component of a modified Floyd-Hoare logic constructed on the base of a composition-nominative approach. According to this approach, Floyd-Hoare assertions are presented with the help of a special composition called Floyd-Hoare composition. Monotonicity and continuity of this composition are proved. The language of the modified Floyd-Hoare logic is described. Further, the inference rules for such logic are studied, their soundness conditions are specified. The logic constructed can be used for program verification.

Keywords. Program algebra, program logic, composition-nominative approach, partial predicate, soundness

Key terms. FormalMethod, VerificationProcess

1 Introduction

Program logics are the main formalisms used for proving assertions about program properties. A well-known Floyd-Hoare logic [1, 2] is an example of such logics. Semantically, this logic is defined for a case of total predicates and functions though programs can be partial. In this case assertions can be presented with the help of a special composition over total predicates and functions called Floyd-Hoare composition (FH-composition). However, a straightforward extension of classical Floyd-Hoare logic for partial predicates and functions meets some difficulties. The first one is that the classical FH-composition will not be monotone. Monotonicity means that the result of the mapping evaluation remains the same on extended data, if it was evaluated on the initial data. This important property grants the possibility to reason about the correctness of the program based on the correctness of its approximations.

That is why the need of a modified definition of the classical Floyd-Hoare logic for the case of partial mappings arises. Here we will consider only mappings (predicates, ordinary functions, and program functions) defined over flat nominative data (nominative sets). Such data are treated as collections of named values. Mappings over such data are called quasiary mappings [3]. The obtained program algebras are called quasiary program algebras. They form a semantic component of quasiary Floyd-Hoare logics.

The syntactic component of such logics is presented by their languages and systems of inference rules. We study the possibility to use classical rules for modified logics with a monotone Floyd-Hoare composition. Systems of such inference rules should be sound and complete to be of a practical use. This could be achieved by adding proper restrictions to the inference rules of the classical Floyd-Hoare logic that fail to be correct. It should be also shown that by weakening additional restrictions we obtain a system of the inference rules that is not sound. This will prove that restrictions are necessary.

The rest of the paper is structured as follows. In Section 2 we describe program algebras of quasiary predicates and functions on different levels of abstraction, define a modified Floyd-Hoare composition and specify the syntax for the modified logic. In Section 3 we prove the main properties of this composition. In Section 4 we study the soundness of the system of inference rules for the introduced program algebras. Finally, we formulate conclusions in Section 5.

2 Quasiary Program Algebras

To modify the classical Floyd-Hoare logic for partial quasiary mappings, we will use semantic-syntactic scheme [3-5]. This means that we will first define the semantics in the form of classes of quasiary program algebras. Then the language of the logic will be defined as well as the interpretation mappings.

To emphasize a mapping's *partiality/totality* we write the sign \xrightarrow{p} for partial mappings and the sign \xrightarrow{t} for total mappings. Given an arbitrary partial mapping

$\mu: D \xrightarrow{p} D', d \in D, S \subseteq D, S' \subseteq D'$ we write:

- $\mu(d) \downarrow$ to denote that μ is defined on d ;
- $\mu(d) \downarrow = d'$ to denote that μ is defined on d with a value d' ;
- $\mu(d) \uparrow$ to denote that μ is undefined on d ;
- $\mu[S] = \{\mu(d) \mid \mu(d) \downarrow, d \in S\}$ to denote the image of S under μ ;
- $\mu^{-1}[S'] = \{d \mid \mu(d) \downarrow, \mu(d) \in S'\}$ to denote the preimage (inverse image) of S' under μ .

2.1 Classes of quasiary mappings

Let V be a set of *names (variables)*. Let A be a set of *basic values*. Given V and A , the class ${}^V A$ of *nominative sets* is defined as the class of all partial mappings from V to A ,

thus, ${}^V A = V \xrightarrow{p} A$. Informally speaking, nominative sets represent states of variables.

Though nominative sets are defined as mappings, we follow mathematical traditions and also use a set-like notation for these objects. In particular, the notation $d = [v_i \mapsto a_i \mid i \in I]$ describes a nominative set d where $v_i \mapsto a_i \in_n d$ means that $d(v_i)$ is defined and its value is a_i ($d(v_i) \Downarrow = a_i$). The main operation for nominative sets is the binary *total overriding operation* $\nabla: {}^V A \times {}^V A \xrightarrow{t} {}^V A$ defined by the formula $d_1 \nabla d_2 = [v \mapsto a \mid v \mapsto a \in_n d_2 \vee (v \mapsto a \in_n d_1 \wedge \neg \exists a' (v \mapsto a' \in_n d_2))]$. Intuitively, given d_1 and d_2 this operation yields a new nominative set which consists of named pairs of d_2 and those pairs of d_1 whose names do not occur in d_2 .

Let $Bool = \{F, T\}$ be the set of Boolean values. Let $Pr^{V,A} = {}^V A \xrightarrow{p} Bool$ be the set of all partial predicates over ${}^V A$. Such predicates are called *partial quasiary predicates*. Let $Fn^{V,A} = {}^V A \xrightarrow{p} A$ be the set of all partial functions from ${}^V A$ to A . Such functions are called *partial quasiary ordinary functions*. Here ‘ordinary’ means that the range of such functions is the set of basic values A . Let $FPrg^{V,A} = {}^V A \xrightarrow{p} {}^V A$ be the set of all partial functions from ${}^V A$ to ${}^V A$. Such functions are called *bi-quasiary functions*.

Quasiary predicates represent conditions which occur in programs, quasiary ordinary functions represent the semantics of program expressions, and bi-quasiary functions represent program semantics.

The terms ‘partial’ and ‘ordinary’ are usually omitted. In a general term, elements from $Pr^{V,A}$, $Fn^{V,A}$, and $FPrg^{V,A}$ are called *quasiary mappings*.

2.2 Hierarchy of program algebras and logics

Based on algebras with three carriers ($Pr^{V,A}$, $Fn^{V,A}$, and $FPrg^{V,A}$) we can define logics of three types:

- *Pure quasiary predicate logics based on algebras with one sort: $Pr^{V,A}$*
- *Quasiary predicate-function logics based on algebras with two sorts: $Pr^{V,A}$ and $Fn^{V,A}$*
- *Quasiary program logics based on algebras with three sorts: $Pr^{V,A}$, $Fn^{V,A}$, and $FPrg^{V,A}$*

For logics of pure quasiary predicates we identify renominative, quantifier, and quantifier-equational levels.

Renominative logics [3] are the most abstract among above-mentioned logics. The main new compositions for these logics are the compositions of renomination (renaming) of the form $R_{x_1, \dots, x_n}^{v_1, \dots, v_n}: Pr^{V,A} \xrightarrow{t} Pr^{V,A}$. Intuitively, given a quasiary predicate p and a nominative set d the value of $R_{x_1, \dots, x_n}^{v_1, \dots, v_n}(p)(d)$ is evaluated in the following way: first, a new nominative set d' is constructed from d by changing the values of the

names v_1, \dots, v_n in d to the values of the names x_1, \dots, x_n respectively; then the predicate p is applied to d' . The obtained value (if it was evaluated) will be the result of $R_{x_1, \dots, x_n}^{v_1, \dots, v_n}(p)(d)$. For this composition we will also use a simplified notation $R_{\bar{x}}^{\bar{v}}$.

The basic compositions of renominative logics are \vee , \neg , and $R_{\bar{x}}^{\bar{v}}$. Note, that renomination (primarily in syntactical aspects) is widely used in classical logic, lambda-calculus, and specification languages like Z-notation, B, TLA, RAISE, ASM, etc.

At the *quantifier* level, all basic values can be used to construct different nominative sets to which quasiary predicates can be applied. This allows one to introduce the compositions of quantification of the form $\exists x$ in style of Kleene's strong quantifiers. The basic compositions of logics of the quantifier level are \vee , \neg , $R_{\bar{x}}^{\bar{v}}$, and $\exists x$.

At the *quantifier-equational* level, new possibilities arise for equating and differentiating values with special 0-ary compositions of the form $=_{xy}$ called equality predicates. Basic compositions of logics of the quantifier-equational level are \vee , \neg , $R_{\bar{x}}^{\bar{v}}$, $\exists x$, and $=_{xy}$.

All specified logics (renominative, quantifier, and quantifier-equational) are based on algebras that have only one sort: a class of quasiary predicates.

For quasiary predicate-function logics we identify the function level and the function-equational level.

At the *function* level, we have extended capabilities for the formation of new arguments of functions and predicates. In this case it is possible to introduce the superposition compositions $S_F^{\bar{v}}$ and $S_P^{\bar{v}}$ (see [4, 5]), which formalize substitution of functions into function and predicate respectively. Also special null-ary denomination parametric compositions (functions) \acute{x} are introduced. The introduction of such functions allows one to model renomination compositions with the help of superpositions. The basic compositions of logics of the function level are \vee , \neg , $S_F^{\bar{v}}$, $S_P^{\bar{v}}$, $\exists x$, and \acute{x} .

At the function-equational level, a special equality composition $=$ can be introduced additionally. The basic compositions of logics of the function-equational level are \vee , \neg , $S_F^{\bar{v}}$, $S_P^{\bar{v}}$, $\exists x$, \acute{x} , and $=$. At this level different classes of first-order logics can be presented.

This means that two-sorted algebras (with sets of predicates and functions as sorts and above-mentioned compositions as operations) form a semantic base for first-order CNL.

The level of *program logics* is quite rich. Investigation of such logics is a special challenge; here we will study semantic properties of a modified *Floyd-Hoare logic*. To define such logics we should first define program algebras with program compositions as their operations. Such compositions correspond to the main structures of programs. In the simplest case they are:

- The parametric assignment composition $AS^x : Fn^{V,A} \rightarrow FPrG^{V,A}$
- The composition of sequential execution $\bullet : FPrG^{V,A} \times FPrG^{V,A} \rightarrow FPrG^{V,A}$
- The conditional composition $IF : Pr^{V,A} \times FPrG^{V,A} \times FPrG^{V,A} \rightarrow FPrG^{V,A}$
- The cyclic composition (loop) $WH : Pr^{V,A} \times FPrG^{V,A} \rightarrow FPrG^{V,A}$

Additionally we need compositions that describe properties of the programs. The Floyd-Hoare composition $FH : Pr^{V,A} \times FPr_g^{V,A} \times Pr^{V,A} \rightarrow Pr^{V,A}$ is the most important of them. Its formal definition will be given in the next subsection.

2.3 Formal definition of a Floyd-Hoare composition

The required definition stems from the treatment of Floyd-Hoare assertions with total predicates (see, for example, [6]). Namely, an assertion $\{p\}f\{q\}$ is said to be valid if and only if

$$\text{for all } d \text{ from } {}^V A \text{ if } p(d) = T, f(d) \Downarrow = d' \text{ for some } d' \text{ then } q(d') = T \tag{1}$$

Note, that we do not make a distinction between a formula and its interpretation. Thus, we treat, say, p as a formula in the assertion $\{p\}f\{q\}$ and as a predicate of the program algebra.

The definition (1) permits to treat $\{p\}f\{q\}$ as a predicate because this is a pointwise definition. Rewriting this definition for different cases we get the following matrices (table 1) specifying the logical values of $\{p\}f\{q\}$ for an arbitrarily d :

Table 1. Logical values of $\{p\}f\{q\}$ for total predicates.

<p>a) $f(d)$ is defined</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 5px;">$p(d) \setminus q(f(d))$</td> <td style="padding: 5px;">F</td> <td style="padding: 5px;">T</td> </tr> <tr> <td style="padding: 5px;">F</td> <td style="padding: 5px;">T</td> <td style="padding: 5px;">T</td> </tr> <tr> <td style="padding: 5px;">T</td> <td style="padding: 5px;">F</td> <td style="padding: 5px;">T</td> </tr> </table>	$p(d) \setminus q(f(d))$	F	T	F	T	T	T	F	T	<p>b) $f(d)$ is undefined</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 5px;">$p(d)$</td> <td style="padding: 5px;">$\{p\}f\{q\}(d)$</td> </tr> <tr> <td style="padding: 5px;">F</td> <td style="padding: 5px;">T</td> </tr> <tr> <td style="padding: 5px;">T</td> <td style="padding: 5px;">T</td> </tr> </table>	$p(d)$	$\{p\}f\{q\}(d)$	F	T	T	T
$p(d) \setminus q(f(d))$	F	T														
F	T	T														
T	F	T														
$p(d)$	$\{p\}f\{q\}(d)$															
F	T															
T	T															

Our aim is to extend the notion of assertion validity for partial predicates. But first we should admit that the presented definition will not be monotone under predicate extension. Indeed, consider informally the following assertion:

$$\{T\} \text{ while } T \text{ do skip } \{F\}.$$

This Floyd-Hoare triple will be true on all data, because the infinite loop is undefined on all data, and thus on all data the condition of validity for this assertion is satisfied. Now consider a triple $\{T\} \text{ skip } \{F\}$ that is false on all data. However, the mapping ‘skip’ is an extension of ‘while T do skip’. Thus, monotonicity fails for a case when $p(d)=T$ and $f(d)$ is undefined. So, the value for this case should be changed.

To define a monotone interpretation of Floyd-Hoare triple for partial predicates we should change the question marks in Table 2 to Boolean values.

Table 2. Logical values of $\{p\}f\{q\}$ for partial predicates, where the question marks represent values that should be changed to proper Boolean values.

$p(d) \setminus q(f(d))$	F	T	$Undefined$
F	T	T	$?$
T	F	T	$?$
$undefined$	$?$	$?$	$?$

$p(d)$	$\{p\}f\{q\}(d)$
F	T
T	$?$
$undefined$	$?$

To define such interpretation we adopt the following requirements:

- Monotonicity of a composition on all its arguments
- Maximal definiteness of the obtained predicates (we call this requirements as Kleene’s principle)

We use techniques for non-deterministic semantics described in [7]. We treat the case when a predicate is ‘undefined’ as non-deterministic values T and F. Thus, we can use matrices from table 1 to evaluate a set of Boolean values for every case. The obtained results are presented in Table 3.

Table 3. Logical values of $\{p\}f\{q\}$ for partial predicates presented as sets of Boolean values.

$p(d) \setminus q(f(d))$	$\{F\}$	$\{T\}$	$\{F, T\}$
$\{F\}$	$\{T\}$	$\{T\}$	$\{T\}$
$\{T\}$	$\{F\}$	$\{T\}$	$\{F, T\}$
$\{F, T\}$	$\{F, T\}$	$\{T\}$	$\{F, T\}$

$p(d)$	$\{p\}f\{q\}(d)$
$\{F\}$	$\{T\}$
$\{T\}$	$\{F, T\}$
$\{F, T\}$	$\{F, T\}$

Now, replacing non-deterministic results $\{F, T\}$ on undefined we get the final results (table 4).

Table 4. Logical values of $\{p\}f\{q\}$ for partial predicates.

$p(d) \setminus q(f(d))$	F	T	$undefined$
F	T	T	T
T	F	T	$undefined$
$undefined$	$undefined$	T	$undefined$

$p(d)$	$\{p\}f\{q\}(d)$
F	T
T	$undefined$
$undefined$	$undefined$

The obtained matrices define an interpretation of $\{p\}f\{q\}$ for partial predicates. As was said earlier, we formalize such triples as a Floyd-Hoare composition $FH : Pr^{V,A} \times FPr^{V,A} \times Pr^{V,A} \rightarrow Pr^{V,A}$ ($p, q \in Pr^{V,A}, f \in FPr^{V,A}, d \in V, A$):

$$FH(p, f, q)(d) = \begin{cases} T, & \text{if } q(f(d)) \downarrow = T \text{ or } p(d) \downarrow = F, \\ F, & \text{if } p(d) \downarrow = T \text{ and } q(f(d)) \downarrow = F, \\ \text{undefined} & \text{in other cases.} \end{cases}$$

2.4 Formal definition of program algebra compositions

In the previous subsection the formal definition of FH-composition was presented. In this subsection we give brief definitions of other compositions (see details in [3-5]).

Propositional compositions are defined as follows ($p, q \in Pr^{V,A}$, $d \in V^A$):

$$(p \vee q)(d) = \begin{cases} T, & \text{if } p(d) \downarrow = T \text{ or } q(d) \downarrow = T, \\ F, & \text{if } p(d) \downarrow = F \text{ and } q(d) \downarrow = F, \\ \text{undefined in other cases.} \end{cases} \quad (\neg p)(d) = \begin{cases} T, & \text{if } p(d) \downarrow = F, \\ F, & \text{if } p(d) \downarrow = T, \\ \text{undefined if } p(d) \uparrow. \end{cases}$$

Unary parametric composition of existential quantification $\exists x$ with the parameter $x \in V$ is defined by the following formula ($p \in Pr^{V,A}$, $d \in V^A$):

$$(\exists x p)(d) = \begin{cases} T, & \text{if } b \in A \text{ exists : } p(d \nabla x \mapsto b) \downarrow = T, \\ F, & p(d \nabla x \mapsto a) \downarrow = F \text{ for each } a \in A, \\ \text{undefined in other cases.} \end{cases}$$

Here $d \nabla x \mapsto a$ is a shorter form for $d \nabla [x \mapsto a]$.

Parametric n-ary superpositions with $\bar{x} = (x_1, \dots, x_n)$ as the parameter are defined by the following formulas ($f, g_1, \dots, g_n \in Fn^{V,A}$, $p \in Pr^{V,A}$, $d \in V^A$):

$$(S_F^{\bar{x}}(f, g_1, \dots, g_n))(d) = f(d \nabla [x_1 \mapsto g_1(d), \dots, x_n \mapsto g_n(d)]),$$

$$(S_P^{\bar{x}}(p, g_1, \dots, g_n))(st) = f(st \nabla [x_1 \mapsto g_1(st), \dots, x_n \mapsto g_n(st)]).$$

Null-ary parametric denomination composition with the parameter $x \in V$ is defined by the following formula ($d \in V^A$): ' $x(d) = d(x)$.

Binary equality composition = is defined as follows ($f, g \in Fn^{V,A}$, $d \in V^A$):

$$(f = g)(d) = \begin{cases} T, & \text{if } f(d) \downarrow, g(d) \downarrow, \text{ and } f(d) = g(d), \\ F, & \text{if } f(d) \downarrow, g(d) \downarrow, \text{ and } f(d) \neq g(d), \\ \text{undefined in other cases.} \end{cases}$$

Identical program composition $id \in FPr^{V,A}$ is the most simple: $id(d) = d$ ($d \in V^A$).

Assignment composition is defined as follows ($f \in Fn^{V,A}$, $d \in V^A$):

$$AS^x(f)(d) = d \nabla [x \mapsto f(d)].$$

Sequential execution is introduced in the ordinary way ($fs_1, fs_2 \in FPr^{V,A}$, $d \in V^A$):

$$fs_1 \bullet fs_2(d) = fs_2(fs_1(d)).$$

Note, that we define \bullet by commuting arguments of conventional functional composition: $fs_1 \bullet fs_2 = fs_2 \circ fs_1$.

Conditional composition depends on the value of the first function which is the condition itself ($p \in Pr^{V,A}$, $fs_1, fs_2 \in FPr^{V,A}$, $d \in V^A$):

$$IF(p, fs_1, fs_2)(d) = \begin{cases} fs_1(d), & \text{if } p(d) \downarrow = T, \\ fs_2(d), & \text{if } p(d) \downarrow = F, \\ \text{undefined in other cases.} \end{cases}$$

Cycle is defined by the following formulas: $WH(p, fs)(d) = d_n$, where $d_0 = d$, $d_1 = fs(d_0)$, ..., $d_n = fs(d_{n-1})$, moreover $p(d_0) \downarrow = T$, $p(d_1) \downarrow = T$, ..., $p(d_{n-1}) \downarrow = T$, ..., $p(d_n) \downarrow = F$ ($p \in Pr^{V,A}$, $fs \in FPr^{V,A}$, $d \in V^A$).

It means that we have defined the following *quasiary program algebra*:

$QPA(V, A) = \langle Pr^{V,A}, Fn^{V,A}, FPr^{V,A}, \vee, \neg, S_F^{\bar{v}}, S_p^{\bar{v}}, 'x, \exists x, =, id, AS^x, \bullet, IF, WH, FH \rangle$.

This algebra is the main object of our investigation.

2.5 Formal definition of program algebra terms

Terms of the algebra $QPA(V, A)$ defined over sets of predicate symbols Ps , function symbols Fs , program symbols Prs , and variables V specify the syntax (the language) of the logic. We now give inductive definitions for terms $Tr(Ps, Fs, Prs, V)$, formulas $Fr(Ps, Fs, Prs, V)$, program texts $Pt(Ps, Fs, Prs, V)$, and Floyd-Hoare assertions $FHFr(Ps, Fs, Prs, V)$.

First we will define terms:

- if $f \in Fs$ then $f \in Tr(Ps, Fs, Prs, V)$
- if $v \in V$ then $'v \in Tr(Ps, Fs, Prs, V)$
- if $f \in Fs$, $t_1, \dots, t_n \in Tr(Ps, Fs, Prs, V)$, and $v_1, \dots, v_n \in V$ are distinct variables then $S_F^{\bar{v}}(f, t_1, \dots, t_n) \in Tr(Ps, Fs, Prs, V)$

Then we will define program texts:

- $id \in Pt(Ps, Fs, Prs, V)$
- if $p \in Prs$ then $p \in Pt(Ps, Fs, Prs, V)$
- if $v \in V$ and $t \in Tr(Ps, Fs, Prs, V)$ then $AS^v(t) \in Pt(Ps, Fs, Prs, V)$
- if $p_1, p_2 \in Pt(Ps, Fs, Prs, V)$ then $p_1 \bullet p_2 \in Pt(Ps, Fs, Prs, V)$
- if $p_1, p_2 \in Pt(Ps, Fs, Prs, V)$ and $b \in Fr(Ps, Fs, Prs, V)$ then $IF(b, p_1, p_2) \in Pt(Ps, Fs, Prs, V)$
- if $p \in Pt(Ps, Fs, Prs, V)$ and $b \in Fr(Ps, Fs, Prs, V)$ then $WH(b, p) \in Pt(Ps, Fs, Prs, V)$

Finally, formulas and Floyd-Hoare triples are defined:

- if $p \in Ps$ then $p \in Fr(Ps, Fs, Prs, V)$
- if $\Phi \in Fr(Ps, Fs, Prs, V)$ then $\neg\Phi \in Fr(Ps, Fs, Prs, V)$
- if $t_1, t_2 \in Tr(Ps, Fs, Prs, V)$ then $t_1 = t_2 \in Fr(Ps, Fs, Prs, V)$
- if $\Phi \in Fr(Ps, Fs, Prs, V)$ and $v \in V$ then $\exists v\Phi \in Fr(Ps, Fs, Prs, V)$
- if $\Phi, \Psi \in Fr(Ps, Fs, Prs, V)$ then $\Phi \vee \Psi \in Fr(Ps, Fs, Prs, V)$;

- if $p \in Ps$, $t_1, \dots, t_n \in Tr(Ps, Fs, Prs, V)$, and $v_1, \dots, v_n \in V$ are distinct variables then $S_p^{\bar{v}}(p, t_1, \dots, t_n) \in Fr(Ps, Fs, Prs, V)$
- if $f \in Pt(Ps, Fs, Prs, V)$ and $p, q \in Fr(Ps, Fs, Prs, V)$ then $\{p\}f\{q\} \in FHFPr(Ps, Fs, Prs, V)$

After syntax and semantics have been defined, we need to specify the interpretation mappings, assuming that interpretation mappings for the predicate symbols $I_{Ps} : Ps \rightarrow Pr^{V,A}$, functional symbols $I_{Fs} : Fs \rightarrow Fn^{V,A}$, and program symbols $I_{Prs} : Prs \rightarrow FPrs^{V,A}$ are given. Let $J_{Fr} : Fr(Fs, Ps, Prs, V) \rightarrow Pr^{V,A}$ denote an interpretation mapping for formulas, $J_{Tr} : Tr(Fs, Ps, Prs, V) \rightarrow Fn^{V,A}$ denote an interpretation mapping for terms and $J_{Pt} : Pt(Fs, Ps, Prs, V) \rightarrow Prs^{V,A}$ denote an interpretation mapping for programs. They are all defined in a natural way, only the case with assertion needs special consideration:

$$J_{FHFPr}(\{p\}f\{q\}) = FH(J_{Fr}(p), J_{Pt}(f), J_{Fr}(q)).$$

An assertion is said to be *valid* (denoted $\models \{p\}f\{q\}$) if a corresponding predicate is *not refutable*.

3 Monotonicity and Continuity of the Floyd-Hoare Composition

In the previous section, a function-theoretic style of composition definitions was used. To prove properties of the FH-composition, it is more convenient to use a set-theoretic style of definition.

The following sets are called respectively *truth*, *false*, and *undefiniteness domains* of the predicate p over D :

$$\begin{aligned} p^T &= \{d \mid p(d) \downarrow = T\}, \\ p^F &= \{d \mid p(d) \downarrow = F\}, \\ p^\perp &= \{d \mid p(d) \uparrow\}. \end{aligned}$$

The following definitions introduce various images and preimages involved in Floyd-Hoare composition:

$$\begin{aligned} q^{-T,f} &= f^{-1}[q^T], \\ q^{-F,f} &= f^{-1}[q^F], \\ q^{-\perp,f} &= f^{-1}[q^\perp], \\ p^{T,f} &= f[p^T], \\ p^{F,f} &= f[p^F], \\ p^{\perp,f} &= f[p^\perp]. \end{aligned}$$

Using these notations we can define FH-composition by describing the truth and false domains of the predicate that is the value of the composition:

$$\begin{aligned} FH(p, f, q)^T &= p^F \cup q^{-T, f}, \\ FH(p, f, q)^F &= p^T \cap q^{-F, f}. \end{aligned}$$

Validity of formulas (predicates) is considered as irrefutability, that is $\models p \Leftrightarrow p^F = \emptyset$. From this follows that

$$\models FH(p, f, q) \Leftrightarrow p^T \cap q^{-F, f} = \emptyset.$$

Let us give a formal definition of the monotone composition.

Composition $C: (FPrg^{V,A})^n \times (Pr^{V,A})^k \times (Fn^{V,A})^m \rightarrow Pr^{V,A}$ is called *monotone* if the following condition holds for all arguments of C :

$$\begin{aligned} f_1 \subseteq g_1, \dots, f_n \subseteq g_n, p_1 \subseteq q_1, \dots, p_k \subseteq q_k, a_1 \subseteq b_1, \dots, a_m \subseteq b_m \Rightarrow \\ C(f_1, \dots, f_n, p_1, \dots, p_k, a_1, \dots, a_m) \subseteq C(g_1, \dots, g_n, q_1, \dots, q_k, b_1, \dots, b_m). \end{aligned}$$

Theorem 1. Floyd-Hoare composition is monotone on every argument.

Let us prove monotonicity on every argument separately, examining their truth and false domains. For truth domain we have:

$$\begin{aligned} p_1 \subseteq p_2 \Rightarrow p_1^T \subseteq p_2^T \Rightarrow p_1^T \cap q^{-F, f} \subseteq p_2^T \cap q^{-F, f} \Rightarrow \\ FH(p_1, f, q)^F \subseteq FH(p_2, f, q)^F. \end{aligned}$$

Similar, for the false domain of the precondition we have:

$$\begin{aligned} p_1 \subseteq p_2 \Rightarrow p_1^F \subseteq p_2^F \Rightarrow p_1^F \cup q^{-T, f} \subseteq p_2^F \cup q^{-T, f} \Rightarrow \\ FH(p_1, f, q)^T \subseteq FH(p_2, f, q)^T. \end{aligned}$$

Thus, $p_1 \subseteq p_2 \Rightarrow FH(p_1, f, q) \subseteq FH(p_2, f, q)$.

In the case of truth domain of postcondition the proof is similar:

$$\begin{aligned} q_1 \subseteq q_2 \Rightarrow q_1^T \subseteq q_2^T \Rightarrow q_1^{-T, f} \subseteq q_2^{-T, f} \Rightarrow p^F \cup q_1^{-T, f} \subseteq p^F \cup q_2^{-T, f} \Rightarrow \\ FH(p, f, q_1)^T \subseteq FH(p, f, q_2)^T. \end{aligned}$$

The same for the false domain of postcondition:

$$\begin{aligned} q_1 \subseteq q_2 \Rightarrow q_1^F \subseteq q_2^F \Rightarrow q_1^{-F, f} \subseteq q_2^{-F, f} \Rightarrow p^T \cap q_1^{-F, f} \subseteq p^T \cap q_2^{-F, f} \Rightarrow \\ FH(p, f, q_1)^F \subseteq FH(p, f, q_2)^F. \end{aligned}$$

Thus, $q_1 \subseteq q_2 \Rightarrow FH(p, f, q_1) \subseteq FH(p, f, q_2)$.

Let us show the monotonicity of the truth domains for the FP-composition:

$$\begin{aligned} f_1 \subseteq f_2 \Rightarrow q^{-T, f_1} \subseteq q^{-T, f_2} \Rightarrow p^F \cup q^{-T, f_1} \subseteq p^F \cup q^{-T, f_2} \\ \Rightarrow FH(p, f_1, q)^T \subseteq FH(p, f_2, q)^T. \end{aligned}$$

Similar, for the false domains:

$$\begin{aligned} f_1 \subseteq f_2 \Rightarrow q^{-F, f_1} \subseteq q^{-F, f_2} \Rightarrow p^T \cap q^{-F, f_1} \subseteq p^T \cap q^{-F, f_2} \Rightarrow \\ FH(p, f_1, q)^F \subseteq FH(p, f_2, q)^F. \end{aligned}$$

Also $f_1 \subseteq f_2 \Rightarrow FH(p, f_1, q) \subseteq FH(p, f_2, q)$.

Thus, it was shown that the composition is monotone on every component, what is needed to be proved.

For the constructed composition even stronger result is true, it is continuous. To show this, the following definitions are made and the notion of continuity is given (see, for example, [6]).

An infinite set of indexed functions (predicates) $\{f_0, f_1, \dots\}, f_i \subseteq f_{i+1}, i \in \omega$ is called a *chain* of functions (predicates).

The *supremum* of the above-mentioned set of indexed functions (predicates) is called *limit* of the chain of functions (predicates), denoted as $\coprod_i f_i$.

The composition $C : (Pr^{V,A})^n \times (Pr^{V,A})^m \times (Fn^{V,A})^l \rightarrow Pr^{V,A}$ is called *continuous* on the first argument if for arbitrary chain $\{f_i | i \in \omega\}$ the following property holds: $C(\coprod_i f_i, g_2, \dots, g_n, p_1, \dots, p_m, q_1, \dots, q_l) = \coprod_i C(f_i, g_2, \dots, g_n, p_1, \dots, p_m, q_1, \dots, q_l)$.

Continuity on the other arguments is defined in a similar manner.

Theorem 2. Floyd-Hoare composition is continuous on every argument.

Though this result follows from the general consideration, we give here its direct proof. Let us show the continuity on the first argument. In the case of other arguments the proof will be similar.

Consider a chain of predicates $\{p_i | i \in \omega\}$. Since Floyd-Hoare composition is monotone, $\{FH(p_i, f, q) | i \in \omega\}$ will also be a chain. We need to show that $FH(\coprod_i p_i, f, q) = \coprod_i FH(p_i, f, q)$.

For the arbitrary data d , there are two different possibilities – $\coprod_i p_i(d) \uparrow$ and $\coprod_i p_i(d) \downarrow$. In the first case none of the elements of the chain is defined on d . Thus $\forall j \in \omega, FH(\coprod_i p_i, f, q)(d) = FH(p_j, f, q)(d)$, therefore needed equality is obvious. If the limit is defined on these data, an element of the chain that is also defined on this data could be found. Otherwise the limit would have been undefined on those data, what is guaranteed by the inclusion relation on the elements of the chain. Let the limit be the element with index k . Then

$$FH(\coprod_i p_i, f, q)(d) = FH(p_k, f, q)(d) \text{ and}$$

$$FH(p_k, f, q)(d) = \coprod_i FH(p_i, f, q)(d),$$

since $\forall i > k, p_i(d) = p_k(d)$ from the definition of the chain.

The following equality is obtained: $FH(\coprod_i p_i, f, q)(d) = \coprod_i FH(p_i, f, q)(d)$.

Since the data was chosen arbitrary, we get $FH(\coprod_i p_i, f, q) = \coprod_i FH(p_i, f, q)$, what was needed to be proved.

The proof for the other arguments (a program and a postcondition) is similar. Thus, it is proven that the monotone Floyd-Hoare composition is also continuous on every argument.

4 Soundness of Inference Rules System in Floyd-Hoare Algebras

In this section we adopt the same convention as earlier that we do not distinguish between syntactic and semantic notation for formulas. We also assume that the algebra $QPA(V, A)$ is fixed and interpretation mappings are also fixed.

Since a result of the Floyd-Hoare composition can be undefined on some data, classical inference rules can be unsound. This informally means that with true preconditions they could give false postconditions. This happens because predicates can be partial and compositions are defined in a way that differs from the classical Floyd-Hoare composition to be monotone. Let us examine the following system of inference rules to find out what conditions are required for rules to be sound:

$$\begin{array}{c}
\{S^{[x]}(p, f)\}AS^x(f)\{p\} \text{ -- } Ax_AS \\
\{p\}id\{p\} \text{ -- } Ax_ID \\
\frac{\{p\}f\{q\}, \{q\}g\{r\}}{\{p\}f \bullet g\{r\}} \text{ -- } Ax_SEQ \\
\frac{\{b \wedge p\}f\{q\}, \{\neg b \wedge p\}g\{q\}}{\{p\}IF(b, f, g)\{q\}} \text{ -- } Ax_IF \\
\frac{\{b \wedge p\}f\{p\}}{\{p\}WH(b, f)\{\neg b \wedge p\}} \text{ -- } Ax_WH \\
\frac{\{p'\}f\{q'\}}{\{p\}f\{q\}} \text{ -- } Ax_CONS
\end{array}$$

Note that we do not include additional conditions for the consequence rule, because in different classes of algebras we will have different conditions.

An assertion $\{p\}f\{q\}$ is said to be *derived* if there exists its derivation tree with rules of the type Ax_AS, Ax_ID on its leaves. Derivability is denoted as $\vdash \{p\}f\{q\}$.

Let us show that for the rules Ax_SEQ, Ax_WH , and Ax_CONS without additional conditions we can give such an example of the application of the inference rule that will have true preconditions and false postconditions.

Consider Ax_SEQ with violation of the condition $p^{T,f} \subseteq q^T$.

If this condition fails then $\exists p, q, f, d : p(d) = T, q(f(d)) \uparrow, \vdash \{p\}f\{q\}$. In this case we will take such r and g that $\vdash \{q\}g\{r\}$ and $g(f(d)) \downarrow, r(g(f(d))) = F$. This is possible if we define them in the following way:

$$g = id, r(x) = \begin{cases} T, x \neq f(d), \\ F, x = f(d). \end{cases}$$

Then $\vdash \{p\}f \bullet g\{r\}$ does not hold, while $p(d) = T$ and $r(f \bullet g(d)) = F$, what is equal to $\{d\} \subseteq p^T \cap r^{-F, f \bullet g}$.

Consider Ax_WH with violation of the condition $(b \wedge p)^{T,f} \subseteq p^T$.

We will construct such b, f , and p that the following properties hold:

$$\begin{aligned} &\models \{b \wedge p\}f\{p\}, (b \wedge p)^{T,f} \not\subseteq p^T, \\ &\models \{p\}WH(b, f)\{\neg b \wedge p\}. \end{aligned}$$

Let $d_1 \neq d_2 \neq d_3$. Then b, f , and p are defined in the following manner:

$$\begin{aligned} b(x) &= \begin{cases} T, x \neq d_3, \\ F, x = d_3. \end{cases} \\ f(x) &= \begin{cases} x, x \neq d_1, d_2, \\ d_2, x = d_1, \\ d_3, x = d_2. \end{cases} \\ p(x) &= \begin{cases} T, x \neq d_2, d_3, \\ \perp, x = d_2, \\ F, x = d_3. \end{cases} \end{aligned}$$

It is not hard to check that the above-mentioned properties are not satisfied:

$$\begin{aligned} &d_2 \in (b \wedge p)^{T,f}, d_2 \notin p^T, \\ &d_1 \in p^T, d_3 = WH(b, f)(d_1), \\ &d_3 \in (\neg b \wedge p)^F \Rightarrow d_1 \in (\neg b \wedge p)^{-F, WH(b, f)}, \\ &d_1 \in (b \wedge p)^T, d_2 \in p^\perp, d_3 \in p^F. \end{aligned}$$

Thus, $\models \{b \wedge p\}f\{p\}$ because for other data p is true.

That proves that the additional condition is necessary because in other cases the rule is not sound while used on such examples.

The case with the rule Ax_CONS is similar to the previous one with the conditions $p^T \subseteq p'^T, q^F \subseteq q'^F$.

So, it was shown that additional conditions are not redundant. Let us show that if additional conditions hold then the rules are sound.

Theorem 3. Inference rules are sound with additional conditions. In other words:

$$\begin{aligned} &\models \{S^{[x]}(p, f)\}AS^x(f)\{p\}, \\ &\models \{p\}id\{p\}, \\ &\models \{p\}f\{q\} \wedge \models \{q\}g\{r\} \wedge p^{T,f} \subseteq q^T \Rightarrow \models \{p\}f \bullet g\{r\}, \\ &\models \{b \wedge p\}f\{q\} \wedge \models \{\neg b \wedge p\}g\{q\} \Rightarrow \models \{p\}IF(b, f, g)\{q\}, \\ &\models \{b \wedge p\}f\{p\} \wedge (b \wedge p)^{T,f} \subseteq p^T \Rightarrow \models \{p\}WH(b, f)\{\neg b \wedge p\}, \\ &\models \{p'\}f\{q'\} \wedge p^T \subseteq p'^T \wedge q^F \subseteq q'^F \Rightarrow \models \{p\}f\{q\}. \end{aligned}$$

Let us prove this for each rule.

For $\models \{S^{[x]}(p, f)\}AS^x(f)\{p\}$ to hold it is needed that the following condition holds: $FH(S^{[x]}(p, f), AS^x(f), p)^F = (S^{[x]}(p, f))^T \cap p^{-F, AS^x(f)} = \emptyset$.

Assume that it is false and the intersection is not empty. Let some data d belongs to the intersection.

If $d \in (S^{[x]}(p, f))^T$ then $p(d \nabla [x \mapsto f(d)]) = T$.

Let $d \in p^{-F, AS^x(f)}$ then $p(AS^x(f)(d)) = p(d \nabla [x \mapsto f(d)]) = F$, what is impossible, thus, the assumption is incorrect and $(S^{[x]}(p, f))^T \cap p^{-F, AS^x(f)} = \emptyset$, similar, $(S^{[x]}(p, f))^T \cap p^\perp = \emptyset$, what means $\models \{S^{[x]}(p, f)\} AS^x(f) \{p\}$.

$\models \{p\} id \{p\}$ follows from the definition.

Let us prove $\models \{p\} f \{q\} \wedge \models \{q\} g \{r\} \wedge p^{T, f} \subseteq q^T \Rightarrow \models \{p\} f \bullet g \{r\}$.

We have $\models \{p\} f \{q\}, \models \{q\} g \{r\}$ that means $p^T \cap q^{-F, f} = \emptyset; q^T \cap r^{-F, g} = \emptyset$. We need to show that $p^T \cap r^{-F, f \bullet g} = \emptyset$.

Let it be false and $\exists d : d \in p^T \wedge d \in r^{-F, f \bullet g}$. This means that $p(d) = T \wedge r(f \bullet g(d)) = F$.

But using the additional condition we have $p^{T, f} \subseteq q^T$, thus $q(f(d)) = T$. That means $f(d) \in q^T$, then $f(d) \notin r^{-F, g}$. This contradicts the fact that $r(f \bullet g(d)) = F \Rightarrow f \bullet g(d) \in r^F \Rightarrow f(d) \in r^{-F, g}$.

We have the contradiction, which means that the assumption is wrong and $p^T \cap r^{-F, f \bullet g} = \emptyset$. Then $\models \{p\} f \bullet g \{r\}$.

Let us prove $\models \{b \wedge p\} f \{q\} \wedge \models \{\neg b \wedge p\} g \{q\} \Rightarrow \models \{p\} IF(b, f, g) \{q\}$.

We have $\models \{b \wedge p\} f \{q\}, \models \{\neg b \wedge p\} g \{q\}$, which means:

$$(b \wedge p)^T \cap q^{-F, f} = \emptyset; (\neg b \wedge p)^T \cap q^{-F, g} = \emptyset.$$

We need to show that $p^T \cap q^{-F, IF(b, f, g)} = \emptyset$.

Let $\exists d : d \in p^T \wedge d \in q^{-F, IF(b, f, g)}$. Then $p(d) = T, q(IF(b, f, g)(d)) = F$.

Let us examine different cases of $b(d)$:

$b(d) \uparrow$ is impossible, because then $IF(b, f, g)(d) \uparrow$ leads to a contradiction with assumptions about existence of such d .

$$b(d) = T \Rightarrow IF(b, f, g)(d) = f(d) \wedge d \in (b \wedge p)^T \wedge IF(b, f, g)(d) \in (b \wedge p)^{T, f}.$$

With properties of the upper part of the inference rule we have:

$$(b \wedge p)^T \cap q^{-F, f} = \emptyset \wedge IF(b, f, g)(d) = f(d) \wedge d \in (b \wedge p)^T \Rightarrow d \notin q^{-F, (IF(b, f, g))}.$$

A case with $b(d) = F$ is similar to the case where $b(d) = T$.

$$(\neg b \wedge p)^T \cap q^{-F, g} = \emptyset \wedge IF(b, f, g)(d) = g(d) \wedge d \in (\neg b \wedge p)^T \Rightarrow d \notin q^{-F, (IF(b, f, g))}.$$

Thus $d \notin q^{-F, (IF(b, f, g))}$ in any case if d is defined which is guaranteed by the assumption. That leads us to the contradiction, so, $p^T \cap q^{-F, IF(b, f, g)} = \emptyset$.

Thus, we have $\models \{p\} IF(b, f, g) \{q\}$.

Let us prove $\models \{b \wedge p\} f \{p\} \wedge (b \wedge p)^{T, f} \subseteq p^T \Rightarrow \models \{p\} WH(b, f) \{\neg b \wedge p\}$.

We have $\models \{b \wedge p\} f \{p\}$, that means: $(b \wedge p)^T \cap p^{-F, f} = \emptyset$.

We need to show that the following condition holds: $p^T \cap (\neg b \wedge p)^{-F, WH(b, f)} = \emptyset$.

Let $\exists d : d \in p^T \wedge d \in (\neg b \wedge p)^{-F, WH(b, f)}$. Then $\exists d_n : d_n = WH(b, f)(d)$, and by the definition of the composition we have $\exists d_1, d_2, \dots, d_n : d = d_1 \wedge d_{i+1} = f(d_i)$, $i = \overline{1, n-1} \wedge b(d_j) = T$, $j = \overline{1, n-1} \wedge p(d_1) = T \wedge b(d_n) = F$ and $(\neg b \wedge p)(d_n) = F$.

Thus, $(b \wedge p)(d_1) = T$. By $(b \wedge p)^{T, f} \subseteq p^T$ we obtain that $p(d_2) = p(f(d_1)) = T$.

Using the induction over a number of loop execution we obtain that $p(d_n) = T$. That means $(\neg b \wedge p)(d_n) = T$. Thus, we obtained contradiction, so, $p^T \cap (\neg b \wedge p)^{-F, WH(b, f)} = \emptyset$.

Let us prove $\models \{p'\}f\{q'\} \wedge p^T \subseteq p'^T \wedge q^F \subseteq q'^F \Rightarrow \models \{p\}f\{q\}$.

We have $\models \{p'\}f\{q'\}$, what means: $p'^T \cap q'^{-F, f} = \emptyset$.

We need to show that $p^T \cap q^{-F, f} = \emptyset$.

Let this condition be false and $\exists d : d \in p^T \wedge d \in q^{-F, f}$. This means that $p(d) = T \wedge q(f(d)) = F$.

By the condition that $p^T \subseteq p'^T$ we obtain $p'(d) = T$.

But $p'^T \cap q'^{-F, f} = \emptyset$, thus, $d \notin q'^{-F, f}$, while $d \in q^{-F, f}$, then $f(d) \downarrow$, and we have $q'(f(d)) \neq F$.

We have a contradiction which means that the assumption does not hold, so, $p^T \cap q^{-F, f} = \emptyset$.

Both conditions are proved, then $\models \{p\}f\{q\}$.

Thus, all rules are inspected and theorem is proved.

Also the condition for the rule Ax_SEQ can be substituted by one of the following: $p^{T, f} \cap q^\perp = \emptyset$, $q^{\perp, g} \cap r^F = \emptyset$ or $q^{F, g} \supseteq r^F$, but none of them is a sufficient condition, because $(\models \{p\}f\{q\} \wedge \models \{q\}g\{r\} \Rightarrow \models \{p\}f \bullet g\{r\}) \Rightarrow p^{T, f} \subseteq q^T$ doesn't hold.

Similar for the rule Ax_WH , the condition could be given in the one of the following manner: $(b \wedge p)^{T, f} \cap p^\perp = \emptyset$, $(b \wedge p)^{\perp, f} \cap p^F = \emptyset$ or $(b \wedge p)^{F, f} \supseteq p^F$, and they are also insufficient.

The conditions for the rule Ax_CONS also are not sufficient. To prove that we need only to show an example when the condition does not hold but the rule does.

But in some cases we can avoid adding the conditions implicitly to the rules.

Theorem 4. For all assertions $\{p\}f\{q\}$ that were inferred using rules of the inference system except Ax_CONS the following properties hold:

$$\begin{aligned} p^{T, f} &\subseteq q^T, \\ p^{T, f} \cap q^\perp &= \emptyset, \\ p^{\perp, f} \cap q^F &= \emptyset, \\ p^{F, f} &\supseteq q^F. \end{aligned}$$

Let us prove the first property by induction. For the fourth property the case is similar and second and third properties are consequences of the first and the fourth respectively.

Induction base: for Ax_ID and Ax_AS proof is obvious.

Induction step. For Ax_SEQ we have:

$$p^{T,f} \subseteq q^T \wedge q^{T,g} \subseteq r^T \Rightarrow p^{T,f \bullet g} \subseteq r^T.$$

The proof of this fact is obvious.

For Ax_IF we need to prove:

$$(b \wedge p)^{T,f} \subseteq q^T \wedge (\neg b \wedge p)^{T,g} \subseteq (q)^T \Rightarrow p^{T,IF(b,f,g)} \subseteq q^T.$$

Consider $d \in p^{T,IF(b,f,g)}$, then $\exists x: (IF(b,f,g)(x) = d) \wedge p(x)$, that leads to two cases:

- $b(x) = T$, then $f(x) \in q^T$, moreover $d = IF(b,f,g)(x) = f(x)$, thus, $d \in q^T$;
- $b(x) = F$, then $g(x) \in q^T$, moreover $d = IF(b,f,g)(x) = g(x)$, thus, $d \in q^T$.

For Ax_WH we need to prove: $(b \wedge p)^{T,f} \subseteq p^T \Rightarrow p^{T,WH(b,f)} \subseteq (\neg b \wedge p)^T$.

Let $d \in p^{T,WH(b,f)}$, then $\exists x: (WH(b,f)(x) = d) \wedge p(x)$, we need to prove that $(\neg b \wedge p)(d) = T$. Let us examine all data that are obtained during the calculation of $WH(b,f)(x)$: $x = x_0$; $x_1 = f(x_0)$... $d = x_n$, $b(x_0) = b(x_1) = \dots b(x_{n-1}) = T$, $b(x_n) = F$, thus from $(b \wedge p)^{T,f} \subseteq p^T$ we have, that $p(d) = p(x_n) = T$, this together with $b(x_n) = F$ gives $(\neg b \wedge p)(d) = T$, what was needed to prove.

The theorem is proved.

Theorem 3 and Theorem 4 together give us the fact that if we declare Ax_CONS in such a way that it retains the properties of the theorem 4, then inference rules system will be sound without addition of new conditions, which will be guaranteed by Theorem 4.

But in this case system would not be complete. Let us give an example.

Let q be an arbitrary predicate that has nonempty truth, false, and undefiniteness domains, and p be such predicate that $p^T = q^T \cup q^\perp$. Then $\models \{p\}id\{q\}$, but $\neg(p^{T,id} \subseteq q^T)$, when the inference rules system was constructed for the following property to hold: $\vdash \{p\}f\{q\} \Rightarrow p^{T,f} \subseteq q^T$.

5 Conclusions

In this paper special program algebras of partial quasiary mappings have been described. Such algebras form a semantic base for a modified Floyd-Hoare logic. In this case assertions have been presented by a special composition called Floyd-Hoare composition. Monotonicity and continuity of this composition have been proved. The language of the modified Floyd-Hoare logic has been described. Further, the inference rules for such a logic have been studied and their soundness conditions have been specified. The logic constructed can be used for program verification.

The major directions of further investigation are the question of completeness of the system of inference rules, invariants for rules, and types for variables and functions. Also the authors plan to construct a prototype of a program system in the style of [8, 9] oriented on the constructed logics.

References

1. Floyd, R. W.: Assigning Meanings to Programs. In: Proc. American Mathematical Society Symposia on Applied Mathematics, vol. 19, pp. 19–31 (1967)
2. Hoare, C. A. R.: An Axiomatic Basis for Computer Programming. *Comm. ACM*, 12, 576–580, 583 (1969)
3. Nikitchenko, M. S., Shkilniak, S. S.: *Mathematical Logic and Theory of Algorithms*. Publishing house of Taras Shevchenko National University of Kyiv, Kyiv (2008) (in Ukrainian)
4. Nikitchenko, M., Tymofieiev, V.: Satisfiability and Validity Problems in Many-Sorted Composition-Nominative Pure Predicate Logics. In: V. Ermolayev et al. (eds.): *ICTERI 2012, CCIS 347*, pp. 89–110. Springer Verlag, Berlin Heidelberg (2013)
5. Nikitchenko, M. S., Tymofieiev, V. G.: Satisfiability in Composition-Nominative Logics. *Central European Journal of Computer Science*, 2(3), 194–213 (2012)
6. Nielson. H.R., Nielson, F.: *Semantics with Applications: A Formal Introduction*. John Wiley & Sons Inc. (1992)
7. Avron, A., Zamanskym A.: Non-Deterministic Semantics for Logical Systems. *Handbook of Philosophical Logic*, vol. 16, pp. 227–304 (2011)
8. Schreiner, W.: Computer-Assisted Program Reasoning Based on a Relational Semantics of Programs. In: P. Quaresma and R.-J. Back (eds.) *Proc 1st Workshop on CTP Components for Educational Software (THedu'11)*, July 31 2011, Wrocław, Poland, No 79 of *Electronic Proceedings in Theoretical Computer Science (EPTCS)*, ISSN: 2075-2180, pp. 124–142 (2012)
9. Schreiner, W.: *A Program Calculus Technical Report*. Research Institute for Symbolic Computation (RISC), Johannes Kepler University, Linz, Austria, <http://www.risc.uni-linz.ac.at/people/schreine/papers/ProgramCalculus2008.pdf> (2008)

A Formal Model of Resource Sharing Conflicts in Multithreaded Java ^{*}

Nadezhda Baklanova and Martin Strecker

Institut de Recherche en Informatique de Toulouse (IRIT), Université de Toulouse

`{nadezhda.baklanova,martin.strecker}@irit.fr`

Abstract. We present a tool for analysing resource sharing conflicts in multithreaded Java programs. We consider two models of execution: purely parallel one and sequential execution on a single processor. A Java program is translated into a system of timed automata which is verified by the model checker UPPAAL. We also present our work in progress on formalisation of Real-Time Java semantics and the semantics of timed automata.

Keywords. resource sharing, Java, timed automata, model checking

Key terms. Model, Development, ConcurrentComputation, FormalMethod, QualityAssuranceProcess

1 Introduction

Along with increasing usage of multithreaded programming, a strong need of sound algorithms arises. The problem is even more important in programming of embedded and real-time systems where liveness conditions are extremely important. To certify that no thread would starve or would be deadlocked, lock-free and wait-free algorithms have been developed. Lock-free algorithms do not use critical sections or locking and allow to avoid thread waiting for getting access to a mutual exclusion object. Nevertheless, only one thread is guaranteed to make progress. Wait-free algorithms prevent starvation by guaranteeing a stronger property: all threads are guaranteed to make progress, eventually. Such algorithms for linked lists, described for example in [4,11], are very complex, difficult to implement and, consequently, hard to verify.

What is worse, these algorithms seem to be incompatible with hard real-time requirements: the progress guarantees are not bounded in time. Thus, a lock-free insertion of an element into a linked list by a thread may need several (possibly infinitely many) retries because the thread can be disturbed by concurrent threads. Under these conditions, it is not possible to predict how much time is needed before the thread succeeds.

^{*} Part of this research has been supported by the project *Verisync* (ANR-10-BLAN-0310)

Critical sections are used in many applications in order to ensure concurrent access to objects although if the scheduling order is wisely planned, locks are not necessary since threads access objects at different moments of time. This is the motivation for our work: we develop a tool for checking resource sharing conflicts in concurrent Java programs based on the statement execution time. This gives a “time-triggered” [6] flavor to our approach of concurrent system design: resource access conflicts are resolved by temporal coordination at system assembly time, rather than during runtime via locking or via retries (as in wait-free algorithms).

We assume that a program is annotated with WCET information known from external sources. The checker translates a Java program into a timed automaton which is then model checked by a tool for timed automata (concretely, UPPAAL).

In this paper, after an informal introduction (Section 2), we present a formal semantics of the components of the translation, namely Timed Automata (Section 3) and a multi-threaded, timed version of Java (Section 4). Then we describe the mechanism of the concrete translator written in OCaml (Section 5) and give some preliminary correctness arguments (Section 6) – the formal verification still remains to be done. The conclusions (Section 7) discuss some restrictions of our current approach and possibilities to lift them.

Status of the present document: We give a glimpse at several aspects of our formalisation, which is far from being coherent. Therefore, this paper is rather a basis for discussion than a finished publication.

2 Informal Overview

To show the main idea, we present an example of a concurrent Java program. It is a primitive producer-consumer buffer with one producer and one consumer where both producer and consumer are invoked periodically. The program is annotated with information about statement execution time in `//@ ... @//` comments.

```
private class Run1 implements Runnable{
  public void run(){
    int value,i;
    //@ 1 @//
    i=0;
    while(i<10){
      synchronized(res){
        //@ 2 @//
        value=Calendar.getInstance().get(Calendar.
          MILLISECOND);
        //@ 5 @//
        res.set(value);
      }
      Thread.sleep(10);
      //@ 2 @//
      i++;
    }
  }
}
```

```

    }
  }
}

private class Run2 implements Runnable{
  public void run(){
    int value,i;
    //@ 1 @//
    i=0;
    Thread.sleep(9);
    while(i<10){
      synchronized(res){
        //@ 4 @//
        value=res.get();
      }
      Thread.sleep(8);
      //@ 1 @//
      i++;
    }
  }
}

```

One of the possible executions is shown in Figure 1.

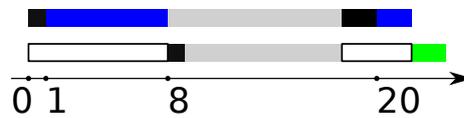


Fig. 1: Possible execution flow. Black areas represent execution without locks, blue and green areas - execution within a critical section, grey areas - sleeping, white areas - waiting for processor time.

After having translated this program to a system of timed automata we run the UPPAAL model checker to determine possible resource sharing conflicts. The checked formula is

$$A[]\forall(i : \text{int}[0, \text{objNumber} - 1])\forall(j : \text{int}[0, \text{autNumber} - 1])\text{waitSet}[i][j] < 1, \quad (1)$$

where *waitSet* is an array of boolean flags indicating whether a thread waits for a lock of a particular object. If all array members in all moments of time are false, no thread waits for a lock therefore no resource sharing conflicts are possible.

3 Timed Automata Model

Timed automata are a common tool for verifying concurrent systems; the underlying theory is described in [1]. We formalize the basic semantics of an extension of timed automata used in the UPPAAL model checker. The formalized syntax and semantics are adapted from [3].

An automaton edge is composed of a starting node, a condition under which the edge may be taken (guard), an action, clocks to reset and a final node.

type-synonym $\langle 'n, 'a \rangle \text{ edge} = 'n \times \text{cconstr} \times 'a \text{ list} \times \text{id set} \times 'n$

An invariant is a condition on a node which must be satisfied when an automaton is in this node.

type-synonym $'n \text{ inv} = 'n \Rightarrow \text{cconstr}$

An automaton consists of a set of nodes, a starting node, a set of edges and an invariant function.

type-synonym $\langle 'n, 'a \rangle \text{ ta} = 'n \text{ set} \times 'n \times \langle 'n, 'a \rangle \text{ edge set} \times 'n \text{ inv}$

We use the model checker UPPAAL which proposes an extension of classical timed automata with variables. A full state comprises a node and a valuation function. There are two types of variables: integer (*aval*) and boolean (*bval*), and clock variables which have a special semantic status in timed automata. The available transitions can be defined knowing given this state.

record *valuation* =
aval:: $\text{id} \Rightarrow \text{nat}$
bval:: $\text{id} \Rightarrow \text{bool}$
cval:: $\text{id} \Rightarrow \text{time}$

type-synonym $'n \text{ state} = 'n \times \text{valuation}$

A timed automaton can perform two types of transitions: timed delay and edge taking. If an automaton takes an edge, variables may be updated or clocks may be reset to 0. The **Transition** constructor takes a list of variable and clock updates as an argument.

datatype $'a \text{ ta-action} = \text{Timestep time} \mid \text{Transition } 'a \text{ list}$

Accordingly, the timed automata semantics has two rules: delay and transition. If an automaton is delayed, it stays in the same node, and values of all clocks are increased to the value d . The invariant of the current node must be satisfied.

If an automaton takes a transition, the node is changed, and clock values remain unchanged unless they are reset to 0. The invariants of both starting and final nodes must be satisfied as well as the guard of the taken transition. If the action of the transition involves variable updates, the valuation function is updated as well. This is done by the function application $\text{eval-stmts varv } a$.

$$\frac{\begin{array}{c} \text{varv}' = \text{varv}(\text{cval} := \text{add}(\text{cval} \text{ varv}) d) \\ \text{varv} \models \text{invs } l \quad \text{varv}' \models \text{invs } l \quad l \in \text{nodes} \end{array}}{(\text{nodes}, \text{init}, \text{edges}, \text{invs}) \vdash (l, \text{varv}) - \text{Timestep } d \rightarrow (l, \text{varv}')}$$

$$\frac{\begin{array}{c} (l, g, a, r, l') \in \text{edges} \\ \text{varv} \models g \quad \text{varv}' = \text{eval-stmts } \text{varv } a(\text{cval} := \text{reset}(\text{cval} \text{ varv}) r) \\ \text{varv}' \models \text{invs } l' \quad l \in \text{nodes} \quad l' \in \text{nodes} \end{array}}{(\text{nodes}, \text{init}, \text{edges}, \text{invs}) \vdash (l, \text{varv}) - \text{Transition } a \rightarrow (l', \text{varv}')}$$

A network of timed automata is a product automaton with exceptions in case of handshaking actions [2]. Handshaking allows to synchronize two automata so that both take an edge simultaneously.

$$\frac{\begin{array}{c} (s_1, g_1, a, cs_1, s_1') \in \text{edges}_1 \quad (s_2, g_2, a, cs_2, s_2') \in \text{edges}_2 \\ a \in \text{getEdgeActions } \text{edges}_1 \cap \text{getEdgeActions } \text{edges}_2 \end{array}}{((s_1, s_2), g_1 \uparrow g_2, a, cs_1 \cup cs_2, s_1', s_2') \in \text{edges-shaking } \text{edges}_1 \text{ edges}_2}$$

$$\frac{\begin{array}{c} (s_1, g, a, cs, s_1') \in \text{edges}_1 \quad a \in \text{getEdgeActions } \text{edges}_1 \\ a \notin \text{getEdgeActions } \text{edges}_2 \quad s_2 \in \text{getEdgeNodes } \text{edges}_2 \end{array}}{((s_1, s_2), g, a, cs, s_1', s_2) \in \text{edges-shaking } \text{edges}_1 \text{ edges}_2}$$

$$\frac{\begin{array}{c} (s_2, g, a, cs, s_2') \in \text{edges}_2 \quad a \in \text{getEdgeActions } \text{edges}_2 \\ a \notin \text{getEdgeActions } \text{edges}_1 \quad s_1 \in \text{getEdgeNodes } \text{edges}_1 \end{array}}{((s_1, s_2), g, a, cs, s_1, s_2') \in \text{edges-shaking } \text{edges}_1 \text{ edges}_2}$$

4 Java Model

The look of the Java semantics has been inspired by the Jinja project [5] and its multithreaded extension JinjaThreads [7]. The Java execution flow is modeled by three transition relations: evaluation, scheduler and platform. The evaluation semantics is the semantics of a single thread, the scheduler semantics is responsible for thread scheduling, and the platform semantics formalizes the notion of time advancement. Taking into account the passage of time is the essential increment wrt. the above-mentioned semantics.

Following Jinja, we do not distinguish expressions and statements; their datatype is the following:

```

datatype expr
= Val val
| Var vname
| VarAssign vname expr
| Cond expr expr expr
| While expr expr

```

| *Annot annot expr*
 | *Sync expr expr*
 | *Sleep expr*

... and others.

The system state is large and complex; it stores local information of all threads such as local variable values and expression to be evaluated, shared objects, time, actions to be carried out by the platform, locks and wait sets. Given this state and scheduler logic, the further execution order is deterministic.

record *full-state* =

threads :: *id* \Rightarrow *schedulable option* — threads pool
th-info :: *thread-id* \Rightarrow *thread-state option* — expression to be evaluated and state
sc-info :: *schedule-info* — locks and thread statuses
pl-info :: *platform-info* — global time
gl-info :: *heap* — heap state
ws-info :: *waitSet* — wait sets
running-th :: *thread-id* — currently running thread
pending-act :: *action* — action to be carried out by platform

Evaluation semantics depends solely on local and heap variables therefore we use the reduced variant of full state for thread evaluation step.

record *eval-state* =

ev-st-heap :: *heap*
ev-st-local :: *locals*

When a thread expression is reduced, the duration of the performed action is not taken into account on the evaluation step, so the action type is passed further to the platform step where time advances according to the action. For now we assume that any action of a particular type takes a fixed amount of time for execution.

The evaluation rules take an expression and a local state and translate them to the new pair of expression and state and also emit an action for the platform step. Here are some examples of evaluation rules.

$$\frac{fs \vdash (e, s) \text{--act--} \rightarrow (e', s')}{fs \vdash (\text{VarAssign } vr \ e, s) \text{--act--} \rightarrow (\text{VarAssign } vr \ e', s')}$$

$$fs \vdash (\text{VarAssign } vr \ (\text{Val } vl), s) \text{--EvalAct } (exec\text{-time } \text{VarAssignAct}) \rightarrow (\text{Val } Unit, s(\llbracket ev\text{-st-local} := ev\text{-st-local } s(vr \mapsto vl) \rrbracket))$$

Several rules use information about locks and wait sets that is not included in the local state therefore they pull it from the full state.

$$\frac{\neg \text{locked } a \ fs}{fs \vdash (\text{Sync } (\text{Val } (\text{Addr } a)) \ e, s) \text{--lock-action } a \ fs \ 0 \rightarrow (\text{Sync } (\text{Val } (\text{Addr } a)) \ e, s)}$$

$$\frac{\text{locked } a \text{ fs} \quad \text{fst } (\text{the } (\text{sc-lk-status } (\text{sc-info } \text{fs}) \text{ a})) \neq \text{runnint-th } \text{fs}}{\text{fs} \vdash (\text{Sync } (\text{Val } (\text{Addr } \text{a})) \text{ e}, \text{ s}) \text{ --lock-action } a \text{ fs } 0 \rightarrow (\text{Throw } [\text{ResourceSharingConflict}], \text{ s})}$$

$$\frac{\text{locked } a \text{ fs}}{\text{fs} \vdash (\text{Sync } (\text{Val } (\text{Addr } \text{a})) (\text{Val } \text{v}), \text{ s}) \text{ --unlock-action } a \text{ fs } 0 \rightarrow (\text{Val } \text{Unit}, \text{ s})}$$

5 Abstracting Java to Timed Automata

We consider two models of program execution. The first one is purely parallel, i.e. each thread is assumed to execute on its own processor so that no thread waits for CPU time. Another model is the sequential one when a program executes on a single processor. The parallel model is easier, however, it does not seem to be realistic. The sequential model represents the realistic situation for real-time Java applications since the RTSJ (Real-Time Specification of Java [8]) specifies the behavior for monoprocessor systems only.

The translated Java programs must be annotated with timing information about execution time of the following statement. The translation uses timing annotations to produce timed automata which model the program. The obtained system is model checked for possible resource sharing conflicts.

5.1 General principles

We suppose that the translated program has a fixed number of threads and shared fields, all of them defined statically. The initialization code for threads and shared fields must be contained in the `main` method. The classes implementing `Runnable` interface must be nested classes in the class containing the `main` method. The required program structure is shown in the figure 2.

Each thread created in the program is translated into one automaton, and one more additional automaton modeling the Java scheduler is added to the generated system.

Java statements are translated into building blocks for condition statement, loop etc. which are assembled to obtain the final automaton. Annotated statement is translated into its own block. Method calls and wait/notify statements are not translated for now.

The timed automata system contains an array of object monitors representing acquired locks on shared objects. When a thread acquires a lock of an object, the monitor corresponding to this object is incremented, and when the lock is released, the monitor is decremented.

There is a number of checks which are performed before on the program source code which guarantee correctness of the generated model. One of the most critical is the requirement that the whole parse tree must be annotated, i.e. for any leaf of the AST there is a timing annotation somewhere above this leaf. With this requirement the behavior of the generated system can be determined in each moment of time.

```

public class Main{
  Res1 field1; //shared fields declaration
  public static void main(String[] args){
    Run1 r1; //declarations of Runnable object instances
    Thread t1,t2; //thread declarations
    r1=new Run1(); //Runnable objects initialization
    field1=new Res1(); //shared fields initialization
    t1=new Thread(null,r1,"t1"); //thread creation
    t1.start(); //thread start
  }
  private class Run1 implements Runnable{
    public void run(){ //thread logic implementation
      ...
    }
  }
}
private class Res1{ //resouce classes
  ...
}

```

Fig. 2: Required program structure

5.2 Parallel model

In the parallel model threads are supposed not to wait for processor time if they want to execute a statement. However, threads can wait for a lock if they need one which has been taken by another thread. Since this model is not very realistic we concentrate on the sequential model further.

5.3 Sequential model

In the sequential model we assume that at every moment of time only one thread or scheduler can execute. Threads which do not execute in a particular moment of time wait for processor time. Also threads can wait for a lock; waiting does not consume CPU time.

Automata communicate with the scheduler through channels: if the scheduler has selected one thread, it sends a message to it so the thread starts executing. After finishing its execution, the thread sends a message to the scheduler, and the next scheduling cycle starts. The scheduler uses channels `run[i]` to call the *i*-th automaton, and the automata use the channel `scheduler` to give the control back to the scheduler.

There is an array of clocks `c[i]`, each of them corresponding to one thread automaton. These clocks are used to calculate time of annotated statements execution or sleeping time. There is also one clock `cGlobal` used for tracking global time.

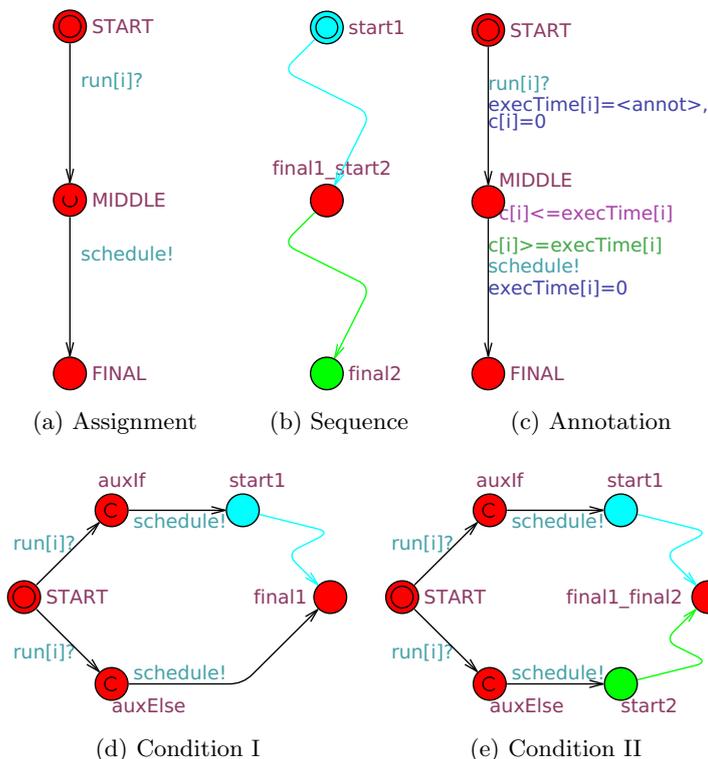


Fig. 3: Building blocks for automata. Elements added on the current step are red; blue and green elements have been generated in the previous step.

The building blocks and their translation are the following for the sequential model:

- (a) Assignment (3a). Three new states and two transitions between them are added. The transition from `START` to `MIDDLE` listens to the channel `run[i]`, and the transition from `MIDDLE` to `FINAL` calls the channel `schedule`. The state `MIDDLE` is urgent since we assume that any statement except the annotated one takes time for execution.
- (b) Sequence (3b). Having two automata with start and final states called `start1`, `start2` and `final1`, `final2` correspondingly, the states `final1` and `start2` are merged.
- (c) Annotation (3c). Three new states and two transition between them are added. The transition from `START` to `MIDDLE` listens to the channel `run[i]`, sets the variable `execTime[i]` to the value of the current annotation and resets the clock `c[i]` to 0. The transition from `MIDDLE` to `FINAL` calls the channel `schedule` and resets the variable `execTime[i]` back to 0. The state `MIDDLE` has an invariant forbidding the automaton to stay in this state if the value of the clock `c[i]` bypasses `execTime[i]`. The transition from `MIDDLE`

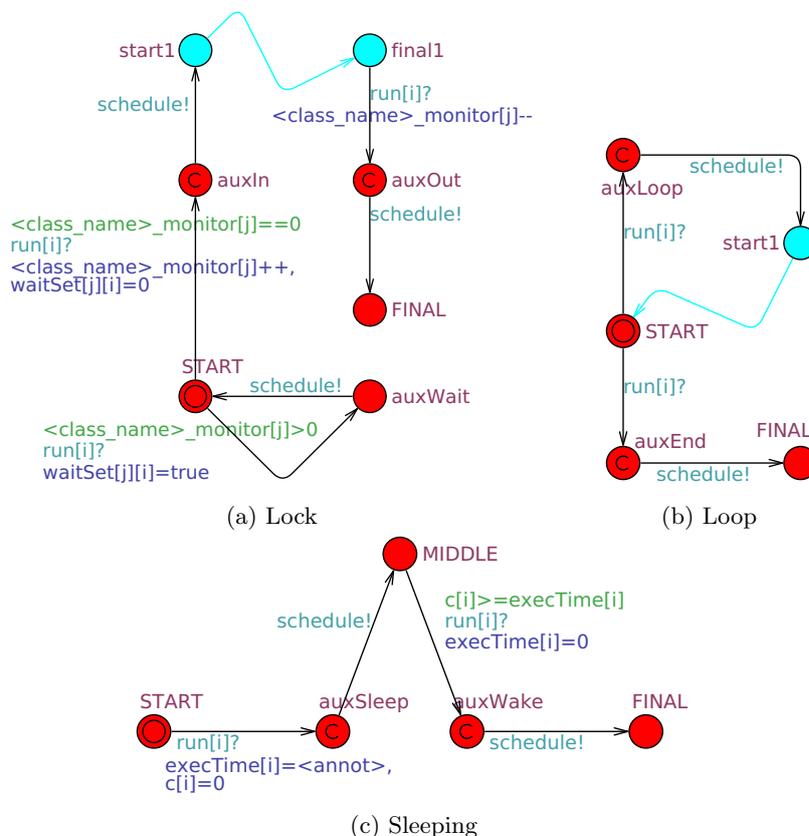


Fig. 4: Building blocks for automata. Elements added on the current step are red; blue and green elements have been generated in the previous step.

- to FINAL has a guard enabling this transition only if the value of $c[i]$ is greater or equal to $execTime[i]$. The invariant and the guard ensure that the automaton would be in the MIDDLE state as long as the annotation claims.
- (d) Condition (3d,3e). Three new states and four transitions are added. If both if and else branches are presented, the final states of automata representing the branch internals are merged. The states **auxIf** and **auxElse** are auxiliary states introduced to divide listening and calling transitions therefore they are made committed. Two transitions from **START** to **auxIf** and **auxElse** listen to the channel $run[i]$, and the transitions from **auxIf** to **start1** and from **auxElse** to **start2** (or to **final1** in case of absence of the else branch) call the channel **schedule**.
- (e) Lock (4a). Two meaningful states and two auxiliary states are added. The transition from **START** listens to the channel $run[i]$ and has a guard checking whether a lock for the object in the argument of the **synchronized** statement is not taken by other threads.

- (f) Loop (4b). Two meaningful states and two auxiliary states are added. One transition from the `START` goes to the next loop iteration, another one exits the loop. Both transitions from `START` to `auxLoop` and `auxEnd` listen to the channel `run[i]`. The transitions from `auxLoop` to `start1` and from `auxEnd` to `FINAL` call the channel `schedule`. Both `auxLoop` and `auxEnd` are made committed. The final state of the automaton corresponding to the loop body is merged with the `START` state.
- (g) Sleeping (4c). The automaton for sleep statement resembles the automaton for annotated statement with additional elements for returning control to the scheduler during sleeping. There are three meaningful and two auxiliary states with transitions connecting them into a chain. The auxiliary states, `auxSleep` and `auxWake`, are committed. The transition from `START` to `auxSleep` listens to the channel `run[i]`, sets the variable `execTime[i]` to the duration of sleeping period and resets the clock `c[i]` to 0. The transition from `auxSleep` to `MIDDLE` calls the channel `schedule` so that the scheduler can schedule other threads. The transition from `MIDDLE` to `auxWake` listens to the channel `run[i]` and has a guard enabling this transition only if `c[i]` is greater or equal to `execTime[i]`. The update on this channel resets the value of `execTime[i]` back to 0. Unlike the automaton for the annotated statement, there is no invariant in the `MIDDLE` state because a thread is not obliged to continue its execution right after it has woken up. It may wait for processor time before. The transition from `auxWake` to `FINAL` calls the `schedule` channel.

5.4 Scheduler

The Java scheduler maintains thread statuses and grants permission to execute to threads. The scheduler model has three states: `scheduling`, `runThread`, `wait`. The scheduler starts in the state `scheduling` which has transitions for updating thread eligibility statuses. When all thread statuses are updated, the scheduler moves to the state `runThread` calling the channel `run[i]` for some thread with index `i` which is eligible for execution. While the thread is executing, the scheduler stays in the state `runThread`. When the thread has finished its execution, it calls a channel `schedule`, and the scheduler returns back to the `scheduling` state, and the new scheduling cycle starts. If there was no thread eligible for execution, the scheduler goes to the `wait` state where it can stay for some time and repeat scheduling.

Each thread gets two transitions for status updates. One assumes that a deadline for an action performing by a thread has passed, another one assumes that the deadline has not been reached yet. In the first case the flag `isEligible[i]` is set to true, and the thread with index `i` can be scheduled for execution. Otherwise, `isEligible[i]` is set to false, and the thread with index `i` cannot be scheduled.

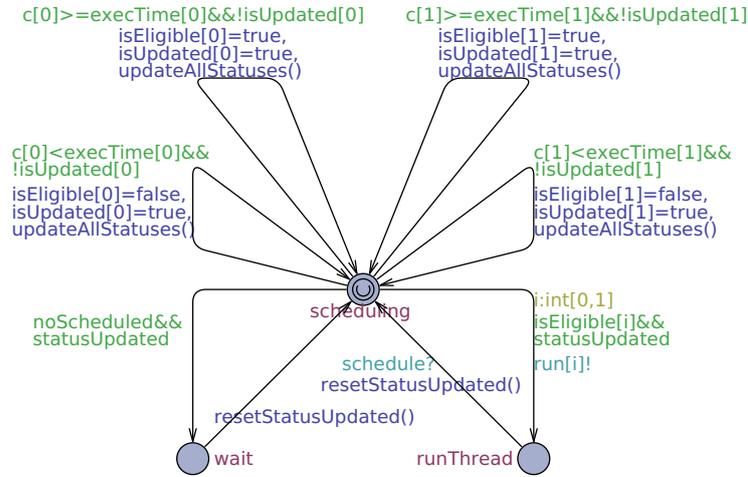


Fig. 5: Scheduler for a system with two threads.

6 Correctness of Abstraction

Java threads have a complex lifecycle shown in Figure 7. We adopt it to our model with limitations (see Figure 6). States in building blocks for automata can be attributed to a single class. For example, the start and end states of any building block belong to the *scheduling* state.

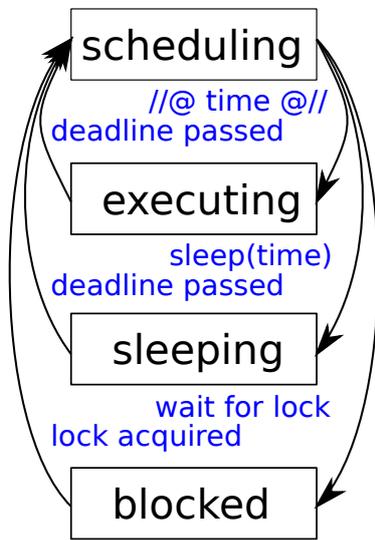


Fig. 6: Thread model used by the translation.

Each thread automaton starts in the *scheduling* state; after being scheduled it can go to one of the three states below. A thread is in the state *executing* if there is an assignment or an annotated statement to be executed. A thread goes to the *sleeping* state if it has met a sleep call; if a thread wants to acquire a lock but it cannot because another thread owns it, the active thread has to go to the *blocked* state and wait until another thread releases the lock.

The listed states are general, and automata may perform several internal steps while staying in the same generalized state.

Based on the state classification above, the simulation function can be easily built by mapping states of Java semantics to states of TA semantics

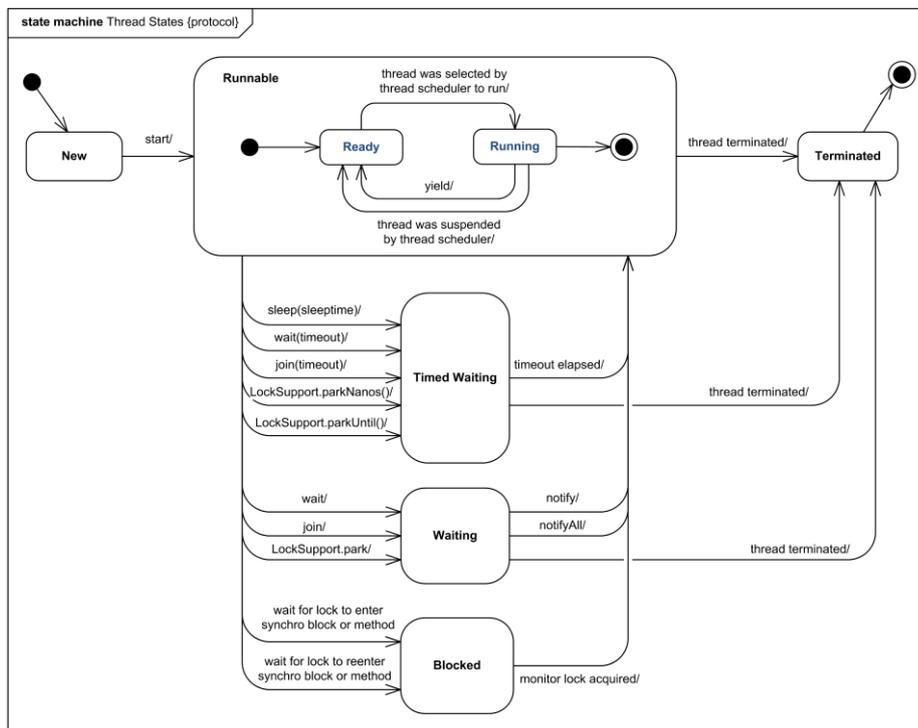


Fig. 7: Java 6 thread model (taken from [12]).

belonging to the same class. This is a preliminary thought concerning a construction of the simulation function.

6.1 Outline of Correctness Proof

We are in the process of carrying out a formal correctness proof. In the foreseeable future, we will concentrate on the following ingredients:

1. We will define a simulation relation \preceq between the TA semantics (Section 3) and the Java execution semantics (Section 4). Both semantics are instances of timed transition systems.
2. We show that for each thread i , the Java execution semantics of thread i is simulated by the execution semantics of the TA obtained from abstracting (Section 5) the thread. Thus: Let J_i be the code executed by thread i . Let $TA_i = \text{abstr}(J_i)$ be the automaton obtained by the abstraction function sketched in Section 5. Then we show that $J_i \preceq TA_i$.
3. We establish an analogous simulation relation between the Java code of the scheduler J_S and the corresponding timed automaton TA_S (Figure 5), and show $J_S \preceq TA_S$.

4. Timed automata can be run in parallel, as sketched at the end of Section 3, by combining them with a parallel composition operator \otimes_{TA} . Similarly, there is a parallel operator \otimes_J for composing thread-local actions (see statement evaluation semantics of Section 4) to obtain a combined transition semantics for the full Java state. We show that these operators are compatible with simulation: $J \preceq TA \wedge J' \preceq TA' \implies J \otimes_J J' \preceq TA \otimes_{TA} TA'$. Combined with the results of (3), we show that our TA abstraction correctly simulates our full Java execution semantics.

Future work will concentrate on establishing a correspondence between timed transition systems (TTSs), logics, and simulation, in the following sense:

1. We will formalize a notion of execution trace of a TTS, formalize an appropriate temporal logic and establish the traditional model relation between traces and formulae of the temporal logic. In this context, we hope to be able to extend previous work described in [9].
2. We will show that simulation \preceq induces a trace inclusion property, which again induces preservation of a certain class of models of formulae.
3. Using this result and the system refinement property of (4), we will show that the correctness formula established by model checking (1) makes indeed a correct prediction about the behaviour of our concurrent Java programs. Differently said: A Java program certified as free of resource access conflicts by model checking has no execution traces in which two threads access a resource at the same time.

7 Conclusions

This paper describes work in progress. The generation of Timed Automata out of Java code is still undergoing major changes, and the precise definition of a correctness statement and its proof still have to be done.

We may however comment on some fundamental assumptions of our approach, which may in part seem unrealistic.

- *Obtaining WCET annotations:* In the examples of this paper, the WCET annotations are fictitious values. There are WCET analysis tools especially geared at Java [10] that could be used to provide these annotations.
- *Worst-case vs. exact execution time:* Even then, the problem remains that our assumptions refer to the exact execution time of the code, whereas a WCET analysis only provides an upper bound. We intend to introduce `sleep` statements at the end of annotation statements so that a thread remains blocked until its WCET has indeed elapsed.
- *Granularity:* The size of code blocks we analyze (included, for example, in annotation statements) is not supposed to be in the order of a few instructions, but in the order of several hundred instructions. This is meant to reduce the relative error when estimating the WCET, and also to obtain reasonably-sized timed automata.

- *Non-interruptible annotation statements*: We presently assume that annotation statements are not interrupted, without verifying it. Future work will try to extend our approach in such a way
 - that threads are annotated with more accurate timing information (such as: periodicity) so that the mentioned assumption can be verified;
 - this assumption can be relaxed and interruption by higher-priority threads is possible, again under the hypothesis that the release parameters of periodic threads are known.

Acknowledgements. We are grateful to Marie Dufлот-Kremer, Pascal Fontaine and Stephan Merz (Loria) and Jan-Georg Smaus (Irit) for discussions about this work.

References

1. Alur, R., Dill, D.L.: A theory of timed automata. *Theoretical Computer Science* 126, 183–235 (1994)
2. Baier, C., Katoen, J.P.: *Principles of Model Checking*. MIT Press (2008)
3. Bengtsson, J., Yi, W.: Timed automata: Semantics, algorithms and tools. In: Desel, J., Reisig, W., Rozenberg, G. (eds.) *Lectures on Concurrency and Petri Nets*, *Lecture Notes in Computer Science*, vol. 3098, pp. 87–124. Springer Berlin / Heidelberg (2004), 10.1007/978-3-540-27755-2
4. Harris, T.L.: A pragmatic implementation of non-blocking linked-lists. In: *Lecture Notes in Computer Science*. pp. 300–314. Springer-Verlag (2001)
5. Klein, G., Nipkow, T.: A machine-checked model for a Java-like language, virtual machine, and compiler. *ACM Trans. Program. Lang. Syst.* 28(4), 619–695 (2006)
6. Kopetz, H., Bauer, G.: The time-triggered architecture. *Proceedings of the IEEE* 91(1), 112–126 (2003)
7. Lochbihler, A.: Verifying a compiler for Java threads. In: Gordon, A.D. (ed.) *European Symposium on Programming (ESOP’10)*. LNCS, vol. 6012, pp. 427–447. Springer (Mar 2010)
8. The Real-Time for Java Expert Group: *The Real-Time Specification for Java* (Jan 2006)
9. Schimpf, A., Merz, S., Smaus, J.G.: Construction of Büchi automata for LTL model checking verified in Isabelle/HOL. In: Nipkow, T., Urban, C. (eds.) *22nd Intl. Conf. Theorem Proving in Higher-Order Logics (TPHOLs 2009)*. *Lecture Notes in Computer Science*, vol. 5674. Springer, Munich, Germany (2009)
10. Schoeberl, M., Pedersen, R.: WCET analysis for a Java processor. In: *Proceedings of the 4th international workshop on Java technologies for real-time and embedded systems*. pp. 202–211. JTRES ’06, ACM, New York, NY, USA (2006)
11. Timnat, S., Braginsky, A., Kogan, A., Petrank, E.: Wait-free linked-lists. In: Ramanujam, J., Sadayappan, P. (eds.) *PPOPP*. pp. 309–310. ACM (2012)
12. [uml-diagrams.org: Java 6 thread states and life cycle. `http://www.uml-diagrams.org/examples/java-6-thread-state-machine-diagram-example.html?context=stm-examples`](http://www.uml-diagrams.org/examples/java-6-thread-state-machine-diagram-example.html?context=stm-examples)

Implementation of Propagation-Based Constraint Solver in IMS

Igor Ol. Blinov

Kherson State University, 27, 40 rokiv Zhovtnya St., Kherson, Ukraine 73000

anubis.igor@gmail.com

Abstract. Article compiling the main ideas of creating propagation-based constraint solver, theoretical basis of constraint programming its implementation in IMS (Insertion Modeling System) and creating prototype of IMS constraint solver.

Keywords. IMS, constraint programming, solvers

Key terms. Mathematical model, process, research

1 Introduction

In today's world, computing problems are becoming more and more applicable in real life. Some problems arise from the real problem of production requirements of science, etc.

The challenge of creating a real-life computational equivalent of the human mind requires that we had better understand at a computational level how natural intelligent systems develop their cognitive and learning functions. The narrow focus of science on this challenge brings together four schools of thought:

1. Computational neuroscience, that tries to understand how the brain works in terms of connectionist models;
2. Cognitive modeling, pursuing higher-level computational description of human cognition;
3. Human-level artificial intelligence, aiming at generally intelligent artifacts that can replace humans at work;
4. Human-like learners: artificial minds that can be understood by humans intuitively, that can learn like humans, from humans and for human needs.

A solution to the problems of this type consists of several parts. The main parts are the tools and methodology. Tools provided at this time to address them varied. We consider the approach of cognitive modeling and cognitive architectures.

Important part of cognitive architecture – solvers. For creating solvers, we use the constraint programming method.

Let's consider this method with its application area and specification.

1.1 Basic Concepts

Constraint programming is a powerful method for solving combinatorial (optimization) problems, which has proven effective and efficient in a wide range of application areas.

Constraint programming is an embedding of constraints in a host language. The first host languages used were logic programming languages, so the field was initially called constraint logic programming. The two paradigms share many important features, like logical variables and backtracking. Today most Prolog (for example) implementations include one or more libraries for constraint logic programming.

The difference between the two is largely in their styles and approaches to modeling the world. Some problems are more natural (and thus, simpler) to write as logic programs, while some are more natural to write as constraint programs.

The constraint programming approach is to search for a state of the world in which a large number of constraints are satisfied at the same time. A problem is typically stated as a state of the world containing a number of unknown variables. The constraint program searches for values for all the variables.

Application areas. Many hard, real-world combinatorial problems lend themselves to modeling as constraint satisfaction or optimization problems. The Handbook of Constraint Programming (Rossi et al., 2006) lists example applications in the areas of scheduling and planning, vehicle routing, configuration, networks (such as power or pipeline networks), and bioinformatics. Further application areas include computational linguistics (for example Duchier, 1999), as well as verification (Yuan et al., 2006) and optimization (van Beek and Wilken, 2001) of computer programs.

Constraint Satisfaction Problems. A combinatorial problem is modeled as a set of variables, representing the objects the problem deals with, and a set of constraints, representing the relationships among the objects. Such a combinatorial problem is called a *Constraint Satisfaction Problem (CSP)*. The common case where the variables can only take values from a finite universe is called a finite domain constraint satisfaction problem. A constraint programming system implements variables and constraints and provides a solution procedure for CSPs, which tries to find an assignment to the variables that satisfies all of the constraints. Clearly, solving CSPs is NP-hard in general, as the satisfiability of Boolean formulas (SAT) is one instance.

As we use the constraint programming approach, solvers are called *constraint solvers*.

Constraint solvers. The success of constraint programming as a field is due to the availability of effective and efficient solution procedures that can solve these practical problems. This paper concentrates on finite-domain constraint programming, implemented in a propagation-based constraint solver, based on exhaustive search. This class of solvers has been successful because of its best-of-several worlds approach. They combine classic AI search methods with advanced implementation techniques from the Programming Languages community and efficient algorithms from Operations Research. Furthermore, the Constraint Programming community has identified global constraints as an important tool to make the structure of constraint problems

explicit and achieve strong propagation. Dedicated propagation algorithms for many different global constraints are available.

We consider the variety of solvers, which are called propagation-based constraint solvers.

Propagation-based constraint solving. At the heart of a propagation-based constraint solver, propagators realize the constraints of a CSP by pruning the variable domains. A propagator removes values from variable domains that cannot be part of any solution of its constraint. Propagators for particular constraints are usually implemented as specialized algorithms. The constraint solver computes a fixed point of all propagators, maximizing the amount of inference they can contribute. It then splits the problem and solves the resulting smaller problems recursively.

This process of inference is called **constraint propagation**. As the main inference method in constraint programming systems, constraint propagation infers that certain values cannot be part of certain variable domains any more because they violate some constraint. The entities that perform constraint propagation are called propagators.

Constraint satisfaction problems are modeled with respect to a finite set of variables X and a finite set of values V . We typically write variables as $x, y, z \in X$, and refer to values as $v, w \in V$.

For article example CSP, we choose the *problem of scheduler generation*. The atom of schedule system is one record of type $\{\text{teacher, group, subject, room}\}$. Let's take this simple set for example. We will describe each part of article using this example, specifying and describing a detail.

1.2 Assignments and Constraints

Constraint satisfaction problem solution should provide a unique correspondence between the values and the variables. A constraint restricts which assignments of values to variables are allowed. Next definition captures assignments and constraints.

Definition 1 An assignment a is a function mapping variables to values. The set of all assignments is $\text{Asn} := X \rightarrow V$. A constraint c is a set of assignments, $c \in \text{Con} := \mathcal{P}(\text{Asn}) = \mathcal{P}(X \rightarrow V)$ (we write $\mathcal{P}(S)$ for the power set of S). It corresponds to a relation over the variables in X . Any assignment $a \in c$ is a solution of c .

In basic works of Guido Tack [1], researchers base constraints on full assignments, defined for all variables in X . However, for typical constraints, only a subset $\text{vars}(c)$ of the variables is significant; the constraint is the full relation for all $x \in \text{vars}(c)$. More formally, a constraint c is the full relation for a variable x if and only:

$$\forall v \in V, \forall a \in c : a[v/x] \in c, \quad (1)$$

where $a[v/x]$ is the assignment a' where $a'(x) = v$ and $a'(y) = a(y)$ for all variables $y \neq x$.

Consequently, the significant variables of c can be defined as

$$\text{vars}(c) := \{x \in X / \exists v \in V; \exists a \in c : a[v/x] \notin c\} \quad (2)$$

Constraints are either written as sets of assignments, or just stated as mathematical expressions with the usual meaning. We use the notation $[[\cdot]]$ when we want to stress that we mean the constraint; for example, we write $[[x < y]]$ to denote the constraint

$\{a \in \text{Asn } a(x) < a(y)\}$.

Using IMS we will define our constraints as a part of function that preceded the insertion function, as we see later. It should done its work before insertion will, or return the forbiddance for insertion. Alternatively, like a part of insertion function, with one more conditional expression.

1.3 Domains and Constraint Satisfaction Problems. Propagators

Constraints represent one half of the Constraint Satisfaction Problem solutions. The other part is the initial set of values that each variable can take. For example in a Golf Club schedule, each variable must take a value from the set of golf-players. In our example, variables will take the value from the set of triples <Teacher, Group, Subject>. A mapping from variables to sets of possible values is a domain. Some popular domains for constraint programming are:

- Boolean domains, where only true/false constraints apply (SAT problem)
- Integer domains, rational domains
- Linear domains, where only linear functions are described and analyzed (although approaches to non-linear problems do exist)
- Finite domains, where constraints are defined over finite sets
- Mixed domains, involving two or more of the above

Finite domains are one of the most successful domains of constraint programming. In some areas (like operation research), constraint programming is often identified with constraint programming over finite domains.

Definition 2 A domain \mathbf{d} is a function mapping variables to sets of values, such that $d(x) \subseteq V$. The set of all domains is $\text{Dom} := X \rightarrow P(V)$. The set of values in \mathbf{d} for a particular variable x , $d(x)$, is called the variable domain of x . A domain \mathbf{d} represents a set of assignments, a constraint, defined as

$$\text{con}(\mathbf{d}) := \{a \in \text{Asn} / \forall x \in X : a(x) \in d(x)\} \quad (3)$$

Said that an assignment $a \in \text{con}(\mathbf{d})$ is licensed by \mathbf{d} .

In our example, we can implement two types of domain realization. Each domain can be realized as a state of an agent, and be (or not) omitted by propagator during insertion, or other way – store all sets of domain in environment' state.

Definition 3 A constraint satisfaction problem (CSP) is a pair $\langle \mathbf{d}, C \rangle$ of a domain \mathbf{d} and a set of constraints C . The constraints C are interpreted as a conjunction of all $c \in C$ and are thus equivalent to the constraint $\{a \in \text{Asn} / \forall c \in C : a \in c\}$. The solutions of a CSP $\langle \mathbf{d}, C \rangle$ are the assignments licensed by \mathbf{d} that satisfy all constraints in C , defined as

$$\text{sol}(\langle \mathbf{d}, C \rangle) := \{a \in \text{con}(\mathbf{d}) / \forall c \in C : a \in c\} \quad)$$

2 Propagators

The basis of a propagation-based constraint solver is a search procedure, which systematically enumerates the assignments licensed by the domain \mathbf{d} of a CSP $\langle \mathbf{d}, C \rangle$.

For each assignment, the solver uses a decision procedure for each constraint to determine whether the assignment is a solution of the CSP. Enumerating all assignments would be infeasible in practice, so in addition to the *decision procedure*, the solver employs a *pruning procedure* for each constraint, which may rule out assignments that are not solutions of the constraint.

Two problems, decision, and pruning procedure for constraints implemented by propagators. Each propagator induces particular constraint. Propagator decides for a given assignment, whether it satisfies the induced constraint, and it can cut off (prune) these tasks from the domain that do not satisfy the constraint. Interleaving propagation and search yield sound and complete procedure for solving CSP. It is complete, because only the assignments that are not solutions are pruned by propagators, and all other assignments are in enum. This is sound, because for each of these tasks, the propagators to decide whether it's the definition of solution. The formal definition of propagators author (see [1]), reflects the minimum properties that are needed in order to get a sound and complete solver. Thus, this model differs from the one commonly found in the literature. Furthermore, knowledge of the unique characteristics of the propagators induced constraints is new. The authors define the propagators in terms domains.

A propagator is a function p that takes a domain as its argument and returns a stronger domain, it may only prune assignments. If the original domain was an assigned domain $\{a\}$, the propagator either accepts it ($p(\{a\}) = \{a\}$) or rejects it ($p(\{a\}) = \emptyset$), realizing the decision procedure for its constraint. In fact, each propagator induces a unique constraint, the set of assignments that it accepts. To make this setup work, we need one additional restriction. The decision procedure and the pruning procedure must be consistent: if the decision procedure accepts an assignment, the pruning procedure must never remove this assignment from any domain—this property is called soundness.

Definition 4 A propagator is a function $p \in Dom \rightarrow Dom$ that is:

- Contracting: $p(d) \subseteq d$ for any domain d
- Sound: for any domain $d \in Dom$ and any assignment $a \in Asn$, if $\{a\} \subseteq d$, then $p(\{a\}) \subseteq p(d)$

The set of all propagators is Prop. If a propagator p returns a strictly stronger domain ($p(d) \subset d$), we say that p prunes the domain d . The propagator p induces the constraint c_p defined by the set of assignments accepted by p :

$$c_p := \{a \in Asn \mid p(\{a\}) = \{a\}\} \quad (5)$$

Soundness expresses exactly that the decision and the pruning procedure realized by a propagator are consistent. A direct consequence is that a propagator never removes assignments that satisfy its induced constraint.

Focusing on our problem, we implement the idea of propagators in additional functions that will proceed the domains (as agent state or environment state) before insertion. Then after the insertion call other propagators to prune from their induced domain unnecessary values to decreasing with each step the search field.

For abstracting the solution of the problem we should give the definition and describing of *propagation problem*, as a higher model of solution of problems of given type.

2.1 Propagation Problem

Propagators were defined as a refinement of constraints – each propagator induces one particular constraint, but in addition has an operational meaning, its pruning procedure. It is possible to define the operational equivalent of a CSP, a propagation problem. Propagation problems realize all constraints of a CSP using propagators.

Definition 5 A propagation problem (PP) is a pair $\langle d, P \rangle$ of a domain d and a set of propagators P . The induced constraint satisfaction problem of a propagation problem $\langle d, P \rangle$ is the CSP $\langle d, \{c_p / p \in P\} \rangle$. The solutions of a PP $\langle d, P \rangle$ are the solutions of the induced CSP, $sol(\langle d, P \rangle) := sol(\langle d, \{c_p / p \in P\} \rangle)$.

The set of solutions of a PP d, P can be defined equivalently as $sol(\langle d, P \rangle) := \{a \in Asn / \forall p \in P : p(\{a\}) = \{a\}\}$, just applying the definitions of induced constraints and solutions of CSPs.

Solution of propagation problem make by using propagators, at each step of inserting an agent into environment. For this, as we mentioned earlier, we will inspect by propagator each domain that is stored in the attributes of the agent. Before each insertion, a domain stored in the attributes of the agent will checked by parameters gained while working. Let's we look at insertion machine architecture

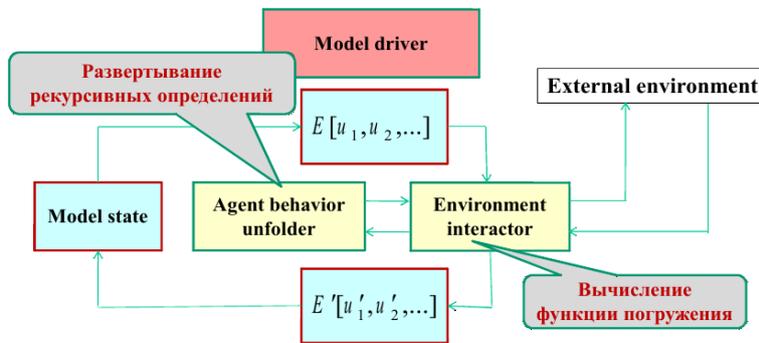


Fig. 1. Insertion machine architecture

The calculation of a insertion function can be tested for the ability to insert the propagator of the agent, which is inserted. If propagator exhausted domain that store in the agent, inserting step will rejected.

Existence of strongest and weakest propagators. Propagators combine a decision procedure with a pruning procedure. While the decision procedure determines the constraint a propagator induces, there is some liberty in the definition of the prun-

ing, as long as it is sound. Thus, there are different propagators for the same constraint, and they can be arranged in a partial order according to their strength:

Definition 6 Let p_1 and p_2 be two propagators that induce the same constraint. Then p_1 is stronger than p_2 (written $p_1 \subseteq p_2$) if and only if for all domains d , $p_1(d) \subseteq p_2(d)$.

2.2 Propagation as a Transition System

Propagation as a Transition System. A propagation-based solver interleaves constraint propagation and search, where constraint propagation means to prune the domain as much as possible using propagators, before search resorts to enumerating the assignments in the domain. Propagating as much as possible means, in the context of propagation problems, to compute a mutual fixed point of all propagators.

Transitions. Let $\langle d, P \rangle$ be a propagation problem. If there is a propagator $p \in P$ that can prune the domain d , that is, if $p(d) \subset d$, then applying p yields a new, simpler propagation problem, $\langle p(d), P \rangle$. Soundness of p makes sure that the new problem has the same set of solutions as the original problem, $sol(\langle d, P \rangle) = sol(\langle p(d), P \rangle)$.

A propagation problem thus induces a transition system, where a transition is possible from a domain d to a domain $d' \subset d$ if there is a propagator $p \in P$ such that $p(d) = d'$. Written such a transition

$$d / -p \rightarrow d' \tag{6}$$

Definition 7 Let d be a domain. A transition $d / -p \rightarrow d'$ with a propagator p to a domain d' is possible if and only if $d' = p(d)$ and $d' \subset d$. The transition system of a propagation problem $\langle d, P \rangle$ consists of all the transitions that are possible with propagators $p \in P$, starting from d . A terminal domain, that is, a domain d such that there is no transition $d / -p \rightarrow p(d)$ for any propagator $p \in P$, is called stable.

Written $d \Rightarrow d'$ if there is a sequence of transitions that transforms d into a stable domain d' . This sequence is empty, $d \Rightarrow d$, if d is stable.

The transition system of a propagation problem is non-deterministic, as there are many possible chains of propagation that result in a stable domain.

Fixed points. The important theorem that ensures that constraint propagation is useful in practice is that, given a propagation problem $\langle d, P \rangle$, its transition system is finite and terminating. No matter in what order the propagators are applied, we reach a stable propagation problem after a finite number of steps.

The naive approach to solving a propagation problem $\langle d, P \rangle$ is to generate all assignments $a \in d$, and then use the propagators $p \in P$ to check whether a satisfies all constraints. This approach makes use of the fact that propagators realize decision procedures for their induced constraints, but does not use their pruning capabilities. A solver that proceeds naively in this fashion is said to follow the *generate-and-test approach*.

3 Conclusion

In conclusion, it can be said that the search for solutions by the *generate-and-test approach* is inefficient, so we will consider other options. Nevertheless, this option works well for prototyping, because of ease of implementation. In the future, we plan to create a working prototype of the university schedule, which plans to make a universal, independent of the input parameters, the types of activities and a list of lessons. The most effective solution to this problem now is supposed to use multi-layer environments, for pruning each of input domains by few environments and few propagators.

References

1. Tack, G.: Constraint Propagation. Models, Techniques, Implementation. Saarbrucken (2009)
2. Letichevsky, A., Letichevskiy, O., Peschanenko, V., Blinov, I., Klionov, D.: Insertion Modeling System and Constraint Programming. In: Ermolayev, V. et al. (eds.) Proc. 7th Int. Conf. ICTERI 2011, Kherson, Ukraine, CEUR-WS vol. 716 (2011)
3. Gilbert, D.R., Letichevsky, A.A.: A Universal Interpreter for Nondeterministic Concurrent Programming Languages. Fifth Compulog Network Area Meeting on Language Design and Semantic Analysis Methods (1996)
4. Letichevsky, A., Gilbert, D.: A General Theory of Action languages. Cybernetics and System Analyses, 1(1), 16–36 (1998)
5. Letichevsky, A., Gilbert, D.: A Model for Interaction of Agents and Environments. In: D. Bert, C. Choppy, P. Moses, (eds.): Recent Trends in Algebraic Development Techniques. LNCS 1827, pp. 311–328, Springer Verlag, Berlin Heidelberg (1999)
6. Letichevsky, A.: Algebra of Behavior Transformations and its Applications. In: Kudryavtsev, V.B., Rosenberg, I.G. (eds) Structural theory of Automata, Semigroups, and Universal Algebra, NATO Science Series II. Mathematics, Physics and Chemistry, Springer, vol. 207, pp. 241–272 (2005)
7. Martin, G., Selic, B. (eds.): UML for Real: Design of Embedded Real-Time Systems. Kluwer Academic Publishers, Amsterdam (2003)
8. Letichevsky, A., Kapitonova, J., Letichevsky, A. Jr., Volkov, V., Baranov, S., Kotlyarov, V., Weigert, T.: Basic Protocols, Message Sequence Charts, and the Verification of Requirements Specifications. Computer Networks, 47, 662–675 (2005)
9. Kapitonova, J., Letichevsky, A., Volkov, V., Weigert, T.: Validation of Embedded Systems. In: Zurawski, R. (ed.) The Embedded Systems Handbook, CRC Press, Miami (2005)
10. Letichevsky, A., Kapitonova, J., Volkov, V., Letichevsky, A. Jr., Baranov, S., Kotlyarov, V., Weigert, T.: System Specification with Basic Protocols. Cybernetics and System Analyses, 4, 479–493 (2005)

UniTESK: Component Model Based Testing

Alexander K. Petrenko¹, Victor Kuliamin¹ and Andrey Maksimov¹

¹ Institute for System Programming of Russian Academy of Sciences

(petrenko, kuliamin, andrew)`@ispras.ru`

Abstract. UniTESK is a testing technology based on formal models or formal specifications of requirements to the behavior of software and hardware components. The most significant applications of UniTESK in industrial projects are described, the experience is summarized, and the prospective directions to the Component Model Based Testing development are estimated.

Keywords. Specification, verification, model-based testing, automated test generation

Key terms. SoftwareSystem, FormalMethod, SpecificationProcess, VerificationProcess

1 Introduction

Model Based testing (MBT) is a rapidly developing domain of software engineering. One of the reasons for such rapid development is the fact that MBT is at the intersection of various other domains of software engineering. In particular, those domains include methods for defining, formalization and modeling of requirements, methods for analysis of both formal specification and formal models as well as the software code, methods for abstraction level control, model transformation and many other software engineering domains. It provides MBT with the ability to quickly adopt the recent achievements proved to be useful in joint domains, in particular, in methods of static analysis and mixed static/dynamic analysis. However, there is no market-ready well-established MBT tool that could be recommended for use in wide range of software development and testing projects. To further develop MBT, we should first analyze the experience gained in the last 15-20 years in this domain. This should help to identify some common problems and focus on their solutions. In this paper, we briefly describe the stages of UniTESK (Unified TEsting & Specification toolKit) development – one of the first MBT tools targeted at testing of wide class of software systems. In the course of the paper, we highlight both positive and negative lessons learned during development and using UniTESK tools. The paper has a subtitle – industrial paper. It means that here we don't reveal any new solutions and don't set

any new scientific problems. We analyze the experience and try to learn lessons that would be useful to researchers working in this domain.

The UniTESK technology [1, 2] was initially developed on the basis of the experience gained in the project on the creation of the automated testing KVEST [3] system, which was developed for testing of the real-time operating system kernel. The work started in 1994 when the term Model Based Testing did not exist. This term appeared at the edge of the 21st century. Currently, MBT is rapidly developed. There are many enthusiasts of this approach and many interpretations of the term itself. To properly position UniTESK in the wide spectrum of MBT solutions, we should first clarify the meaning of MBT within the UniTESK framework.

The following definition is currently given in Wikipedia: “Model-based testing is application of Model based design for designing and optionally also executing artifacts to perform software testing. Models can be used to represent the desired behavior of a System Under Test (SUT), or to represent testing strategies and a test environment”.

This definition includes almost all known interpretations of this term. However, most researchers and practitioners mean more specific approaches and testing techniques by MBT. The first main dividing line is the choice of the modeled object: some model the behavior of the target system (SUT), others model the environment of the target system, in particular, the test itself or the testing system, which, of course, are external to the target system. In UniTESK, the model specifies the system behavior. There are also various types of MBT in this approach, which differ in the way of the behavior description. About the first type Jan Paleska [4] says: “the behavior of the system under test (SUT) is specified by a model elaborated in the same style as a model serving for development purposes”. Such specifications or models are called *executable*. The role of the executable model can be played either by prototype implementation algorithm or by some model which explicitly contains the notion of calculation/execution, for example, finite-state machine, Petri net, ASM [5], etc. Examples of other types of MBT, i.e. “*nonexecutable*”, are algebraic specifications, software contracts in the form of pre- and post-conditions of functions. Each of the model types has its own advantages and drawbacks when testing different SUTs. Besides, when generating tests, not only test data and the sequence of calls of the tested functions should be generated. Also required are the “artifacts” mentioned in the Wikipedia definition, for example, test oracles – the components of the test suite that automatically evaluate the results of the SUT execution whether they meet the requirements or not. Executable models often do not allow test oracle creation, but they are very good for generation of test sequences. Software contracts simplify generation of test oracles, but they do not allow effective generation of test sequences. In other words, several types of models required for generation of effective test suites. Back to UniTESK, we can say that the main model type in it is the software contract in the form of pre- and post-conditions of the functions. In addition, online construction of finite-state machine is used in UniTESK making possible the generation of rather non-trivial test sequences.

UniTESK is a technology that can be implemented on various software platforms and, therefore, can be used for testing of API in various programming languages.

Currently, the most actively used implementations of UniTESK are for C, C++, Java, Python. The corresponding tools are: CTESK, C++TESK, JavaTESK, and PyTESK [6].

UniTESK is an academic product developed in ISPRAS. The UniTESK tools are available under the free license. Experience of industrial application of UniTESK is fused into the tools. Some test suites developed with UniTESK are included in official test suites for certification of industrial software. For example, the OLVER test suite [7] is one of the biggest MBT test suites in the world and yields to the only test suite developed within the framework of the Microsoft Interoperability Initiative [8].

2 UniTESK Usage Review

Let's consider the most interesting examples of UniTESK application and experience gained in them.

The first application of UniTESK was in the project supported by Microsoft Research on development of the MBT test suite for IPv6 implementation [9]. The project started in 2000. At that time UniTESK was at the beginning of its development, so a simplified (light) implementation of API testing in C was used – CTESK-light. In spite of the tool instability, it allowed creation of the effective test suite that detected defects, which were not detected by other test suites. It was the first experience of using contract specifications for telecommunication protocol testing. It was demonstrated that contract specifications in combination with the technique of testing systems with asynchronous interfaces developed within UniTESK [10, 11] allow creating effective tests (they detected more defects, consumed less space and required less effort for development and maintenance than tests developed with traditional technologies). However, the experience of protocols testing showed that besides the post-conditions of the functions in the form of predicates it is useful to have executable models when testing protocols.

One of the first experiences of using UniTESK implementation for testing Java API [12] was the project on testing of Java run-time infrastructure developed as an alternative to the popular Java-platforms. The development of the models and the tests was not a problem since the interfaces were well documented. In addition to Java interfaces, the target system contained also the interfaces in C++, but they also were not a big problem since UniTESK architecture provides the layer of mediators-adapters. The problems revealed when the actual testing started. MBT test suite with online test generation is a fairly complicated program that has strong requirements to the execution platform. In this case, the SUT itself was the execution platform which still was not stable at that time. As a result, the test suite indicated the presence of defects “everywhere”, which, in turn, was of no help to the developers.

The significant application of UniTESK on Java platform (JavaTESK) is the project on testing of infrastructure of the distributed information system of one of Russian major mobile telephony provider. This project is still in progress. The possibility of formal and rigorous specification of the components interfaces became the main advantage of UniTESK for the customer in comparison with the other tools for func-

tional testing. Hundreds of components were formally specified and tested with UniTESK. By the end of the first year of using UniTESK the positive effect appeared in shorter time of integration of new versions of the distributed system. However, a serious problem revealed. In the previous UniTESK applications the requirements to most interfaces were defined by standards and other well-developed documents. But here the level of components documentation often appeared to be insufficient for creation of consistent specifications. Recovery of documentation or requirements to interfaces in the systems of such size becomes almost unsolvable task, which often makes it impossible to use MBT in corpore. Possible solution of this problem will be briefly discussed in Conclusion.

The largest example of UniTESK application is the OLVER (Open Linux VERification) project [7] fulfilled in 2005-2007 under support of the Russian Ministry of Education and Science. The goal of the project was to create formal specifications of interfaces defined in the Linux Standard Base (LSB) standard or in LSB Core – the central part of this standard, to be more exact. The LSB Core includes the most important libraries of OS Linux which implement most of the POSIX standard. The rigorous description of the LSB standard and the test suite capable of a high-quality checking of conformance of any Linux library implementation to the requirements of the standard is a very powerful tool for providing portability of OS Linux applications from one Linux distribution to another. The portability problem is very critical in the Linux ecosystem, since several hundreds of very different distributions are available. The project results are open [7]. The contract specifications of more than 1500 interfaces in C were created. Naturally, the CTEST tool was used for modeling and test generation. In this project, the problems in the standards were also revealed: in LSB (ISO/IEC 23360) and in The Single UNIX Specification containing the POSIX.1 standard (aka IEEE Std 1003.1, aka ISO/IEC 9945, aka The Open Group Base Specifications Issue 6) as its significant part. The developed test suite is included into the package of the certification tests of the international consortium The Linux Foundation [13].

The experience of interface formalization for a large industrial standard and test suite development for such standard gave many lessons to learn. One of such lessons is importance of informational and methodological organization of such project. The amount of documentation and sources, especially with respect to multiple versions and variants for different hardware platforms, is huge. Besides, the development of the standard and development of interface implementations involve thousands of people around the world. It means that the documentation maintenance and availability is one of the most important concerns of the projects of such scale. On the organizational and methodical side, we faced the fact that the training of new employees and the specification and tests quality control require a lot of effort, and the quick achievement of the required professional level is still impossible. In other words, the scalability of the MBT projects in the part of increasing the number of specification and verification experts is one of the most complicated problems preventing MBT from wide introduction.

One of the methodical problems is the choice of the abstraction level for the model. More abstract models or models separated into two-three layers of different abstrac-

tion levels simplify the reuse of the models and tests yielding, however, the bigger and more complex test system. In the long term, it's better to have multilayer models, while in the short term the models close to implementation in the detail level (of course, if the implementation already exists) are more appropriate. A professional and experienced verification expert can find the balance between the abstract description of the behavior, for example, of a file system and specifics and details of the interface of its particular implementation. UniTESK provides special support for the separation of abstraction levels. In particular, the specifics of interfaces can be encapsulated in the mediator-adapter layer. The choice of the balance is determined by the long-term plans on using and improvement of the models and the test suite. So, the work of such kind requires a broad experience and long-term planning skills, which can hardly be expected from ordinary test engineers.

The results of the OLVER project were used later on in the development of the test suite for the Russian real-time operating system OS2000/3000 [14]. This system provides two groups of interfaces. The first group meets the requirements of the POSIX standard, the second one – the requirements of the ARINC-653 international standard for the embedded and other safety critical systems. The definition of the adapter layer separating model and implementation representations of the interfaces provided by the UniTESK architecture significantly simplified the OLVER reuse in this project.

Along with the start of the OLVER project, the work on the UniTESK application to testing of microprocessor designs [15] has been started. Hardware units being parts of Russian microprocessors with the MIPS architecture and microprocessors with VLIM/EPIC elements became the systems under test in this case. The size of typical units in such microprocessors is several millions of gates. The tools required no significant modifications for specification and test generation since CTESK was used as the basis. Technically, binding CTESK to corresponding API of microprocessor model simulator is not a problem, because most simulators that work with modeling languages for microprocessors logic (HLD – High Level Design languages), for example, VHDL or Verilog, provide suitable interface to C programs. Pre-conditions semantics in contract specifications had to be slightly modified. They now describe not just the domain of input data, but rather the operation execution readiness conditions in the given time frame. The same as in the case of protocols modeling, the use of explicit models of the target device behavior (functionality) along with the post-conditions in the form of predicates appeared to be necessary.

Similar to the projects on verification of software systems, one of the main problems preventing MBT from introduction into practice (as well as many other verification methods) is the lack of documentation and other descriptions of functional requirements to components. However, the situation in microprocessors development is slightly better than in the case of software development, because in this case it is customary to build system and architectural models of instruction set semantics along with the HLD models. Elements of these architectural models can be used to fill the gap in the knowledge on behavior of some microprocessor design units [16]. It appears also relatively simple to implement parallel test execution on clusters. Typical size of the finite-state machine generated during test execution for one microprocessor unit is millions of nodes and dozens of millions of transitions. The algorithm of FSM

generation and exploration on clusters with up to 200 nodes appeared to require just 10-15% overhead, i.e. scalability coefficient is close to 1.

It is important to mention the verification tasks that, on the one hand, could not be reduced to modeling with contract specifications, and, on the other hand, pushed forward the development of new MBT methods. In the first place, the task of compiler testing should be mentioned, as well as the task of testing microprocessor as a whole, the so-called “core testing”. The both cases are the tasks of system testing, where test data and test stimuli are submitted to a big “black box” (in our case, these are test programs submitted to the compiler or loaded into memory of the microprocessor simulator), and it is interesting to test not just everything, but some specific behavior modes or specific group of units. In the case of compiler testing, the OTK tool has been developed that was used for testing of optimizing Intel compilers and Simulink [17, 18]. It allows targeting on specific kinds of optimizations. In the case of microprocessor design verification, the MicroTESK tool [19, 20, 21] was developed. The main goal of this tool is checking of various situations appearing in the most complicated subsystems of memory control: TLB, cache and Memory Management Unit (MMU) as a whole.

3 Conclusions and Further Work

Let's start with positive conclusions.

3.1 Positive Conclusions on Modern State of Using MBT

- The world experience is confirmed [22], MBT can be effectively used in industrial projects, and in comparison with the traditional testing MBT gives a unique advantage – many defects can be found in requirements, which are often much more expensive than the defects in implementation
- The achievable level of test coverage is significantly higher than the traditional one (even in comparison with the “white box” testing). Thus, in the case of using OTK for testing GCC compiler, the achieved test coverage was 95%, and in the case of the Intel compiler this level was 75% that was significantly higher than the level achieved by traditional tests [18].
- Although the multi-level structure of specifications (several levels of abstraction) is seldom used in practice, the explicit separation of adapters layer simplifies testing and maintenance and, vice versa, the lack of the corresponding level of adaptation makes test suite development significantly more complicated, which was demonstrated in the Microsoft Interoperability Initiative program [23]
- Online generation of test sequences with the FSM exploration method can be efficiently parallelized and allows using computational resources of clusters with just 10-15% overhead, at least in the case of microprocessor models testing
- The demand of MBT in the safety critical area increases. This tendency can be found in standards defining requirements to development processes for safety critical systems, for example, in DO178C [24] and in Common Criteria [25].

3.2 Negative Aspects of the Modern State in the MBT Area

- The main obstacle preventing MBT from wide introduction into practice is the absence of specifications/models in casual software development. That is, the lack of specifications is often not only the consequence of insufficient attention to specification development or the consequence of short resources. The main reason is often the lack of qualified specialists who are experts in the knowledge domain and at the same time can create specification/model necessary for test generation.
- If MBT is used in projects that do not involve MDD (Model Driven Development) approach, then the model development delays the appearance of first tests – this does not allow obtain tests early in the development. If MBT is used within MDD, then the problems still remain, because different models required for development and for testing, in particular, for generation of different artifacts of the test suite. It is often considered as unacceptable additional cost, while with proper planning many components of the development models can be reused during test generation as demonstrated, for example, in M. M. Chupilko paper [16].
- Bilingual test generation systems like UniTESK and first versions of SpecExplorer [26], specification notations even close to conventional programming languages, for example, JML [27] make deployment of such systems difficult. Bilingual notations require special training of the staff and need permanent and expensive maintenance. Still note that the modern object-oriented languages already have advanced means for writing specifications just in the same language [26-31].

3.3 Directions of Further Works

- A variety of modeling paradigms should be used in various project contexts, in particular, contract specifications, various types of executable models, for example, finite-state machines, Kripke structures, etc. [32]. It is not obvious that the transformation of models from one paradigm into another one will bring real benefit. Each of the model kinds is suitable for analysis of specific aspects of the system behavior, so we should not expect that, for example, a functional model will facilitate estimation of the execution time and memory required. However, obtaining some skeleton or a prototype of the model of one kind on the basis of another kind is quite possible.
- The development of various tools for modeling and specification description for the MBT purposes is required. In spite of the progress in the area of technologies for development of Domain Specific Languages (DSL), practically, the systems based on universal languages benefit from the large number of programmers knowing such languages. The same can be also said about monolingual systems – they overtake multilingual ones.
- Modern achievements in the area of static and hybrid static-dynamic analysis allow integration of these techniques into the MBT systems, at that, the models/specifications as well as software implementations should be the subject of this analysis.

- To overcome the problems with the extreme lack of specifications in real practice, tools for work with requirements and models (see, for example, [33]), in particular, with system models [34, 35] should be developed and deployed. For multicomponent systems, MBT tools should be integrated with the tools for architecture and process mining.

References

1. Bourdonov, I. B., Kossatchev, A. S., Kuli Amin, V. V., Petrenko, A. K.: UniTesK Test Suite Architecture. In: FME 2002. LNCS 2391, pp. 77–88. Springer-Verlag (2002)
2. Kuli Amin, V. V., Petrenko, A. K., Kossatchev, A. S., Bourdonov, I. B.: The UniTesK Approach to Designing Test Suites. *Programming and Computer Software*, 29(6), 310–322 (2003)
3. Bourdonov, I. B., Kossatchev, A. S., Petrenko, A. K., Galter, D.: KVEST: Automated Generation of Test Suites from Formal Specifications. In: *Proceedings of Formal Method Congress, Toulouse, France, 1999*. LNCS 1708, pp. 608–621 (1999)
4. Peleska, J.: Industrial-Strength Model-Based Testing – State of the Art and Current Challenges. Invited Talk. In: Petrenko, A. K., Schlingloff, H. (eds.) *Proceedings Eighth Workshop on Model-Based Testing (MBT 2013)*, Rome, Italy, 17th March 2013. *Electronic Proceedings in Theoretical Computer Science*, 111, pp. 3–28. DOI: 10.4204/EPTCS.111.1 (2013)
5. Börger, E., Stärk, R.: *Abstract State Machines: A Method for High-Level System Design and Analysis*. Springer-Verlag (2003)
6. UniTESK technology, <http://unitesk.ispras.ru>
7. OLVER project, <http://linuxtesting.org>
8. Microsoft Interoperability Initiative, <http://www.microsoft.com/openspecifications>
9. Pakulin, N. V., Khoroshilov, A. V.: Development of Formal Models and Conformance Testing for Systems with Asynchronous Interfaces and Telecommunications Protocols. *Programming and Computer Software*, 33 (6), 316–335 (2007)
10. Khoroshilov, A. V.: *Specification and Testing of Components with Asynchronous Interfaces*. Candidate's thesis, Moscow (2006)
11. Kuli Amin, V. V., Petrenko, A. K., Pakulin, N. V.: Extended Design-by-Contract Approach to Specification and Conformance Testing of Distributed Software. In: *Proceedings of WMSC'2005, Orlando, USA, July 10-13, 2005*. *Model Based Development and Testing*, v. VII, pp. 65–70 (2005)
12. Bourdonov, I. B., Demakov, A. V., Jarov, A. A., Kossatchev, A. S., Kuli Amin, V. V., Petrenko, A. K., Zelenov, S. V.: Java Specification Extension for Automated Test Development. In: *Proceedings of PSI'01*. LNCS 2244, pp. 301–307. Springer-Verlag (2001)
13. The Linux Foundation Consortium. LSB Certification Test Suite, http://ispras.linuxbase.org/index.php/LSB_Certification_System
14. Maksimov, A. V.: Requirements-Based Conformance Testing of ARINC 653 Real-Time Operating Systems. In: *Proceedings of the Data Systems in Aerospace (DASIA 2010) Conference, 2010*. ESA SP-682, ISBN 978-92-9221-246-9 (2010)
15. Ivannikov, V. P., Kamkin, A. S., Kossatchev, A. S., Kuli Amin, V. V., and Petrenko, A. K.: The Use of Contract Specifications for Representing Requirements and for Functional Testing of Hardware Models. *Programming and Computer Software*, 33(5), 272–282 (2007)

16. Chupilko, M. M.: Developing Test Systems of Multi-Modules Hardware Designs. ISSN 0361-7688, Programming and Computer Software, 38(1), 34–42, Pleiades Publishing, Ltd. (2012)
17. Zelenov, S. V., Zelenova, S. A.: Model-Based Testing of Optimizing Compilers. In: Proc. of the 19th IFIP TC6/WG6.1 International Conference on Testing of Software and Communicating Systems – 7th International Workshop on Formal Approaches to Testing of Software (TestCom/FATES 2007). LNCS 4581, pp. 365–377. Springer-Verlag, Berlin Heidelberg (2007)
18. Zelenov, S. V., Silakov, D. V., Petrenko, A. K., Conrad, M., Fey I.: Automatic Test Generation for Model-Based code Generators. In: IEEE ISoLA 2006 Second Intern. Symposium on Leveraging Applications of Formal Methods, Verification and Validation. Paphos, Cyprus, pp. 68-75 (2006)
19. Kamkin, A. S.: A method of Automation of Simulation Testing of Microprocessors with Conveyor Architecture Basing on Formal Specifications. Candidate's thesis, Moscow (2009)
20. Kornyxhin, E. V.: A Method of Automation of Testing Program Generation for MMU Verification. Candidate's thesis, Moscow (2010)
21. Kamkin, A.S., Tatarnikov, A.: MicroTESK: An ADL-Based Reconfigurable Test Program Generator for Microprocessors. In: Proceedings of the 6th Spring/Summer Young Researchers' Colloquium on Software Engineering (SYRCoSE 2012), May 30-31, 2012, Perm, Russia (2012)
22. MBT Survey, <http://www.robertvbinder.com/docs/arts/MBT-User-Survey.pdf>
23. Grieskamp, W.: Microsoft's Protocol Documentation Program: A Success Story for Model-Based Testing. In: Testing – Practice and Research Techniques. LNCS 6303, p. 7 (2010)
24. Adams, C.: Safety-Critical Software for Mission-Critical Applications to Get Boost with Release of DO-178C. Military & Aerospace Electronics, 10 (2010)
25. Common Criteria, <http://www.commoncriteriaportal.org>
26. SpecExplorer, <http://research.microsoft.com/en-us/projects/specexplorer>
27. The Java Modelling Language (JML), <http://www.eecs.ucf.edu/~leavens/JML/index.shtml>
28. Pakulin, N. V.: Integrated Modular Avionics: New Challenges for MBT. In: ETSI TTCN-3 User Conference and Model Based Testing Workshop, Bangalore, India, 11-14 June 2012 (2012)
29. Code Contracts, <http://research.microsoft.com/en-us/projects/contracts>
30. C++TESK, <http://forge.ispras.ru/projects/cpptesk-toolkit>
31. Kuli Amin, V. V.: Component Architecture of Model-Based Testing Environment. Programming and Computer Software, 36(5), 289–305 (2010)
32. Kuli Amin, V. V.: Multi-paradigm Models as Source for Automated Test Construction. In: Proceedings of the 1-st Workshop on Model Based Testing (MBT'2004, in ETAPS'2004), Barcelona, Spain, March 27-38, 2004, Electronic Notes in Theoretical Computer Science, 111:137-160, Elsevier, (2005)
33. ReQuality tool, <http://requality.org/en/doc.en.html>
34. Khoroshilov, A. V., Albitskiy, D., Koverninskiy, I. V., Olshanskiy, M. Yu., Petrenko, A. K., Ugnenko, A. A.: AADL-Based Toolset for IMA System Design and Integration. SAE Int. J. Aerosp. 5(2) (2012)
35. Systems Modeling Language (SysML), <http://www.sysml.org>

Protoautomata as Models of Systems with Data Accumulation

Irina Mikhailova¹ and Boris Novikov² and Grygoriy Zholtkevych²

¹ Luhansk Taras Shevchenko National University,
Institute of Information Technology, 2, Oboronna Str., 91011, Luhansk, Ukraine

`mia_irina@rambler.ru`

² V.N. Karazin Kharkiv National University,
School of Mathematics and Mechanics, 4, Svobody Sqr., 61022, Kharkiv, Ukraine

`{bvnovikov46,g.zholtkevych}@gmail.com`

Abstract. In the paper formal models of software systems and their components based on the notion of an abstract machine are discussed. Necessity to model systems with data accumulation sets the problem of study of generalizations of the notion of an abstract automaton. In the paper two generalizations, namely, preautomata and protoautomata, are considered. It is shown that passing from automata via preautomata to protoautomata can be naturally realized using the language and methods of category theory.

Keywords. system modelling, abstract automaton, category of automata, preautomata, category of preautomata, globalization, protoautomaton, category of protoautomaton, reflector, free protoautomaton

Key terms. MathematicalModel, SpecificationProcess, VerificationProcess

1 Introduction

Theory of abstract state machines or abstract automata is widely applied in different areas of Computer Science. While the early applications of automata theory were connected with theory of compilers design (see, for example, [1]), the more recent its applications are focused on the problems of specification and verification of behaviour of software components [3, 11]. Such changing of the object of the theory was marked by R. Milner in [11]: “In the classical theory, rather little attention is paid to the way in which two automata may interact, in the sense that an action by one entails a complementary action by another. This kind of interaction requires us to look at automata in new light; in particular, this interdependency of automata via their actions seems to demand a new approach to behavioural equivalence”.

But the practice of modelling system behaviour based on the automata approach has shown that the approach is inadequate if data accumulation for the correct response is necessary.

Using the concept of partial action of a semigroup on a set [7, 10], we have defined the notion of preautomaton and studied its properties [4, 12]. The further study has shown that preautomata can be used for modelling some aspects of behaviour of systems with a delayed response [13, 14].

In this paper, we consider a more general class of automaton-liked systems — the class of protoautomata. All necessary information from the theory of semigroups, automata theory, and category theory can be found in the monographs [5, 6, 8, 9].

We use the notation $\varphi : A \dashrightarrow B$ for the partial mapping of A to B (unlike the complete mapping $A \rightarrow B$). If $\varphi(a)$ is not defined for $a \in A$, we write $\varphi(a) = \emptyset$. The free monoid on the alphabet Σ is denoted by Σ^* , and its unit by ε . All actions and preactions used in the paper are right, as it is common in the automata theory.

2 Preliminaries

We will use the definition of the automaton in the following form (the condition of the finiteness is ignored):

Definition 1. *Given a set X and a free monoid Σ^* over the alphabet Σ , an **automaton** is a mapping $X \times \Sigma^* \rightarrow X : (x, a) \mapsto xa$ such that for all $x \in X$ and $u, v \in \Sigma^*$*

$$x\varepsilon = x, \quad (1)$$

$$x(uv) = (xu)v. \quad (2)$$

More general concept is the following

Definition 2 (see [4]). *A **preautomaton** is such a partial mapping of $X \times \Sigma^* \dashrightarrow X : (x, a) \mapsto xa$, that*

- a) *the condition (1) is fulfilled;*
- b) *if $xu \neq \emptyset$ and $(xu)v \neq \emptyset$, then $x(uv) \neq \emptyset$ and equality (2) is fulfilled;*
- c) *if $xu \neq \emptyset$ and $x(uv) \neq \emptyset$, then $(xu)v \neq \emptyset$ and equality (2) is fulfilled.*

The preautomata over the monoid Σ^* form a category $\mathcal{PAut}(\Sigma)$; its morphisms are such maps $\varphi : X \rightarrow Y$ that

$$(\forall a \in \Sigma^*)(\forall x \in X)(xa \neq \emptyset \implies \emptyset \neq \varphi(x)a = \varphi(xa)). \quad (3)$$

The category $\mathcal{Aut}(\Sigma)$ of the automata over Σ is a full subcategory of $\mathcal{PAut}(\Sigma)$.

Preautomata appear in the following situation. Let Y be an automaton and X an arbitrary nonempty subset of Y . Then a restriction of an action on X is a preautomaton.

Conversely, let $X \times M \dashrightarrow X$ be a preautomaton. The construction which is inverse to restriction is called globalization. More precisely:

Definition 3. A **globalization** of the preautomaton X is an automaton Z with an injection $\iota : X \rightarrow Z$ such that for all $a \in \Sigma^*$, $x \in X$

$$\begin{aligned} xa \neq \emptyset &\implies \emptyset \neq \iota(x)a = \iota(xa), \\ \iota(x)a \in \iota(X) &\implies xa \neq \emptyset \ \& \ \iota(xa) = \iota(x)a. \end{aligned}$$

Obviously, ι is a morphism of $\mathcal{P}Aut(M)$. We also call it a globalization.

Definition 4. A globalization $\iota : X \rightarrow Z$ is called **universal** if for any globalization $\iota' : X \rightarrow Z'$ there is a unique morphism $\varkappa : Z \rightarrow Z'$ such that $\iota' = \varkappa\iota$.

The following construction gives an universal globalization (obviously unique up to isomorphism) for any preautomaton $X \times \Sigma^* \dashrightarrow X$. Define a relation \vdash on the set $X \times \Sigma^*$:

$$(x, ab) \vdash (xa, b) \iff xa \neq \emptyset. \tag{4}$$

Let \simeq be an equivalence relation generated by \vdash , and $X^U = (X \times \Sigma^*) / \simeq$. An equivalence class of \simeq containing a pair (x, a) is denoted by $[x, a]$. For $[x, a] \in X^U$ and $b \in \Sigma^*$, we set $[x, a]b = [x, ab]$. Thus a complete action on X^U is defined.

Theorem 1. The automaton X^U with a morphism $\iota^U : X \rightarrow X^U : x \mapsto [x, \varepsilon]$ is the universal globalization of the preautomaton X .

Proof. See [4, Theorem 2] □

3 Protoautomata

The main object of this paper is a generalization of the notion of preautomaton:

Definition 5. A **protoautomaton** is a partial mapping $X \times \Sigma^* \dashrightarrow X : (x, a) \mapsto xa$ such that

- a) the condition (1) is fulfilled;
- b) if $xu \neq \emptyset$ and $(xu)v \neq \emptyset$, then $x(uv) \neq \emptyset$ and equality (2) is fulfilled.

We will also denote the protoautomaton from this definition simply by X , if it does not cause a confusion.

Example 1. Let S be a free subsemigroup of Σ^* and $\alpha : X \times S \rightarrow X$ an automaton. Define a partial mapping $X \times \Sigma^* \dashrightarrow X$ as an extension of α , putting $xu = \emptyset$ for $u \in \Sigma^* \setminus S$; so we get a protoautomaton over Σ^* . Note that in general it is not a preautomaton. In addition, this example shows that the automaton over an infinite alphabet can be represented as a protoautomaton over a two-letter alphabet.

Example 2. Let $X = \{x, y\}$ be a two-element set, L a subset of Σ^* . Define a protoautomaton $X \times \Sigma^* \dashrightarrow X$ putting for $a \neq \varepsilon$

$$xa = \begin{cases} y, & \text{if } a \in L, \\ \emptyset, & \text{if } a \notin L, \end{cases}$$

and $ya = \emptyset$. This example shows that protoautomata recognize all languages.

We denote the category of protoautomata with morphisms defined by the condition (3) by $\mathcal{PtAut}(\Sigma)$; clearly, $\mathcal{PAut}(\Sigma)$ is its subcategory.

It follows from the theory of partial action of semigroups [6, Theorem 5.7], that a protoautomaton which is not a preautomaton has no globalization. More precisely, for the protoautomaton X we can construct an automaton X^U as in Sec. 2, but in this case the morphism ι^U is not injective in general.

In this situation, the concept of a reflector is useful. We recall its definition [9]:

Definition 6. A subcategory \mathcal{D} of a category \mathcal{C} is called **reflective** if with each object $C \in \mathcal{C}$ an object $R_{\mathcal{D}}(C) \in \mathcal{D}$ is associated (called **D-reflector** of the object C) and a morphism $\rho_{\mathcal{D}}(C) : C \rightarrow R_{\mathcal{D}}(C)$ (**reflection morphism**) such that for each $D \in \mathcal{D}$ the diagram

$$\begin{array}{ccc} C & \xrightarrow{\rho_{\mathcal{D}}(C)} & R_{\mathcal{D}}(C) \\ \downarrow & & \\ D & & \end{array}$$

can be extended uniquely to a commutative diagram by some morphism out $\text{Hom}_{\mathcal{D}}(R_{\mathcal{D}}(C), D)$.

It is convenient to use another description of the equivalence \simeq :

Lemma 1. Define a relation $\#$ on the set $X \times \Sigma^*$:

$$(x, a) \# (y, b) \iff (\exists a', b', p \in \Sigma^*)(a = a'p \ \& \ b = b'p \ \& \ xa' = yb' \neq \emptyset).$$

Let \approx be the equivalence relation generated by $\#$. Then \approx coincides with \simeq .

Proof. If $(x, a) \# (y, b)$ then

$$(x, a) = (x, a'p) \vdash (xa', p) = (yb', p) \dashv (y, b'p) = (y, b),$$

whence $\approx \subseteq \vdash \simeq$.

Conversely, if $(x, a) \vdash (y, b)$, then $a = cb, y = xc$ for some $c \in \Sigma^*$. Hence $\vdash \subseteq \#$. Consequently, $\approx \supseteq \simeq$ □

Remark 1. Obviously, \vdash is reflexive and transitive, while $\#$ is reflexive and symmetric.

Lemma 2. Let X be a protoautomaton, Y be a preautomaton (both over Σ^*), $\alpha : X \rightarrow Y$ be a morphism, $x, y \in X$, $a \in \Sigma^*$. Then $[x, \varepsilon] = [y, a]$ implies $\alpha(x) = \alpha(y)a \neq \emptyset$.

Proof. It follows from the condition that

$$(x, \varepsilon) \# (z_1, b_1) \# \dots \# (z_n, b_n) \# (y, a)$$

for some $z_1, \dots, z_n \in X$, $b_1, \dots, b_n \in \Sigma^*$.

Apply induction on n . Since $x = z_1 b_1 \neq \emptyset$ then $\alpha(x) = \alpha(z_1) b_1$. Suppose that $\alpha(x) = \alpha(z_n) b_n \neq \emptyset$. By definition of the relation $\#$

$$b_n = cp, \quad a = dp, \quad z_n c = yd \neq \emptyset$$

for some $c, d, p \in \Sigma^*$. Then $\alpha(z_n) c \neq \emptyset$ and by the induction $\alpha(z_n)(cp) \neq \emptyset$. Since Y is a preautomaton then

$$\alpha(x) = \alpha(z_n)(cp) = \alpha(z_n c)p = \alpha(yd)p = (\alpha(y)d)p = \alpha(y)a.$$

Proof has completed □

Similarly (and even easier) one can prove

Lemma 3. *Let X be a protoautomaton, Y be an automaton (both over Σ^*), $\alpha : X \rightarrow Y$ be a morphism, $x, y \in X$, $a, b \in \Sigma^*$. Then $[x, a] = [y, b]$ implies $\alpha(x)a = \alpha(y)b$.*

Proof is omitted □

We set $[X, \varepsilon] = \{[x, \varepsilon] \in X^U \mid x \in X\}$. Obviously, $[X, \varepsilon]$, being a subset of X^U , is a preautomaton, and in addition, $\iota^U(X) = [X, \varepsilon]$.

Theorem 2. *Let X be a protoautomaton over Σ^* , then*

1. $[X, \varepsilon]$ is a reflector for X in the category $\mathcal{P}Aut(\Sigma)$,
2. X^U is a reflector for X in $Aut(\Sigma)$,
3. X^U is a reflector for $[X, \varepsilon]$ in $Aut(\Sigma)$.

Proof. 1) Let Y be some preautomaton and $\alpha : X \rightarrow Y$ be a morphism of protoautomata. The required morphism $\beta : [X, \varepsilon] \rightarrow Y$ is uniquely determined from the equality $\alpha = \beta \iota^U$. Indeed, for $x \in X$ we have $\alpha(x) = \beta \iota^U(x) = \beta([x, \varepsilon])$. It follows from Lemma 2 that β is well-defined.

2) Similarly, using Lemma 3.

3) Follows from 1), 2), and the following well-known fact [9]:

If $A \subset B \subset C$ are categories, A is reflective in B , and B is reflective in C , then A is reflective in C . Moreover, the reflection morphism from C to A is the product of the corresponding reflection morphisms from C to B and from B to A □

Corollary 1. *$Aut(\Sigma)$ is a reflective subcategory of $\mathcal{P}Aut(\Sigma)$. Moreover, the universal globalization of a preautomaton is its reflector.*

Example 3. Let $X = \{x, y, z, t\}$, $p, u, v \in \Sigma^* \setminus \{\varepsilon\}$. We set $zu = zv = t$, $z(up) = x$, $z(vp) = y$ and $sw = \emptyset$ for all $s \in X$, $w \in \Sigma^* \setminus \{\varepsilon, p, u, v\}$. In such a manner X turns into a protoautomaton. Since $(x, \varepsilon) \# (z, up) \# (z, vp) \# (y, \varepsilon)$ then $[x, \varepsilon] = [y, \varepsilon]$ and the reflection morphism of X is non-injective.

A large class of protoautomata is contained in the following example.

Example 4. Consider a preautomaton $X \times \Sigma^* \dashrightarrow X$ as a directed weighted multigraph with states as vertices and with edges of the form (x, u, y) , where $x, y \in X$, $u \in \Sigma^*$, and $y = xu$. Let U be an arbitrary subset of edges of X . Build a transitive closure U^t of the set U , extending it step by step by the rule: if the edges (x, u, y) and (y, v, z) are at some stage in the expansion, then on the next step we include the edge (x, uv, z) . Then U^t is a protoautomaton.

Example 3 shows that there exists a protoautomaton such that it can not be embedded into some preautomaton (and thus into some automaton).

4 Free Protoautomata

It is well known [2] that free automata play a significant role in the theory of automata (for example, in the problem of constructing a minimal realization). Therefore, it is advisable to consider the question about the existence of free objects in the category of protoautomata.

Recall the necessary definitions:

Definition 7. Let \mathbf{C} and \mathbf{D} be categories, $F : \mathbf{C} \rightsquigarrow \mathbf{D}$ be a functor, C be an object of \mathbf{C} . An object D of \mathbf{D} is called **free on C with respect to the functor F** , if there is a morphism $\alpha : C \rightarrow FD$ such that for any object $D' \in \mathbf{D}$ and any morphism $\beta : C \rightarrow FD'$ there exists the unique morphism $\gamma : D \rightarrow D'$ such that

$$F(\gamma)\alpha = \beta. \quad (5)$$

We consider a category $\mathcal{R}el(\Sigma)$ whose objects are pairs (X, ρ) , where X is a set ($X \in \mathit{Set}$), $\rho \subset X \times \Sigma^*$ is a binary relation such that $X \times \{\varepsilon\} \subset \rho$. A morphism $\phi : (X, \rho) \rightarrow (Y, \sigma)$ of $\mathcal{R}el(\Sigma)$ is a map $\phi : X \rightarrow Y$ such that $(\phi x, u) \in \sigma$ for $(x, u) \in \rho$.

Next, let F be a forgetful functor $F : \mathcal{P}tAut(\Sigma) \rightsquigarrow \mathcal{R}el(\Sigma)$ mapping each protoautomaton $X \times \Sigma^* \dashrightarrow X$ to the pair (X, ρ) with $\rho = \{(x, u) \mid xu \neq \emptyset\}$.

Theorem 3. For each object $(X, \rho) \in \mathcal{R}el(\Sigma)$ there is a protoautomaton that is free on it with respect to the forgetful functor F .

Proof. For $(X, \rho) \in \mathcal{R}el(\Sigma)$ construct a protoautomaton $M = (\rho \times \Sigma^* \dashrightarrow \rho)$, defining the action by the rule

$$(x, u)v = \begin{cases} (x, uv), & \text{if } (x, uv) \in \rho \\ \emptyset, & \text{if } (x, uv) \notin \rho. \end{cases}$$

Then $FM = (\rho, \hat{\rho})$, where $\hat{\rho} = \{(x, u, v) \mid (x, u)v = (x, uv)\} \subset \rho \times \Sigma^*$. Define the morphism $\alpha : (X, \rho) \rightarrow (\rho, \hat{\rho})$ by the formula $\alpha(x) = (x, \varepsilon)$.

Let us show that M is a free protoautomaton on (X, ρ) .

Let $N = (Y \times \Sigma^* \dashrightarrow Y)$ be some protoautomaton and $FY = (Y, \sigma)$. For the required morphism $\gamma : M \rightarrow N$ of (5) we have:

$$\gamma(x, \varepsilon) = F(\gamma)(x, \varepsilon) = F(\gamma)\alpha(x) = \beta(x).$$

Then for any $u \in \Sigma^*$ one can obtain

$$\gamma(x, u) = \gamma(x, \varepsilon)u = \beta(x)u,$$

i.e. γ is uniquely determined □

5 Conclusion

It seems that the class of protoautomata, which has been introduced in the paper, gives the most abstract models for systems with discrete behaviour. This class of abstract machines includes not only machines reacting on the received data immediately, as automata, but it also includes machines whose reactions depend on the accumulated information.

The machines of this class having a greedy behaviour are united into a subclass whose instances are called preautomata. Machines of the subclass are used for modelling behaviour systems for complex event processing as it was shown earlier [13, 14]. This class of machines, in contrast to the class of automata, is closed under structural decomposition, and hence, is more suitable for specifying complex systems. But the condition c) in the definition of a preautomaton (see Definition 2) seems unnatural. This condition also impedes definition of a nondeterministic preautomaton.

Therefore, by eliminating the condition c) we provide a possibility to study nondeterministic models. In our opinion, the models derived in this way (protoautomata) are interesting objects that can be used for specification and verification of complex systems.

References

1. Aho, A.V., Ullman, J.D.: Theory of Parsing, Translation, and Compiling. Prentice-Hall, New York (1972)
2. Arbib, M.A., Manes, E.G.: Machines in a category: an expository introduction. *SIAM Rev.* 16, 163–192 (1974).
3. Börger, E., Stärk, R.: Abstract State Machines: A Method for High-Level System Design and Analysis. Springer-Verlag, Berlin Heidelberg (2003)
4. Dokuchaev, M., Novikov, B., Zholtkevych, G.: Partial actions and automata. *Alg. and Discr. Math.* Vol. 11, 2, 51–63 (2011)
5. Eilenberg, S.: Automata, Languages, and Machines, vol. B. Academic Press, New York (1976)
6. Holcombe, W.M.L.: Algebraic Automata Theory. Cambridge Univ. Press (1982)
7. Hollings, C.: Partial actions of monoids. *Semigroup Forum.* 75, 293–316 (2007)
8. Lallement, G.: Semigroups and combinatorial applications. John Wiley, New York (1979)
9. MacLane, S.: Categories for the Working Mathematician. Springer, Berlin (1971)
10. Megrelishvili, M., Schröder, L.: Globalization of confluent partial actions on topological and metric spaces. *Topol. and Appl.* 145, 119–145 (2004)
11. Milner, R.: Communicating and Mobile Systems: The Pi Calculus. Cambridge University Press, Cambridge (1999)

12. Novikov, B., Perepelytsya, I., Zholtkevych, G. Pre-automata as mathematical models of event flows recognisers. In: V. Ermolayev et al. (eds.) Proc. 7-th Int. Conf. ICTERI 2011, 41–50 (2011)
13. Perepelytsya, I., Zholtkevych, G.: On some class of mathematical models for static analysis of critical-mission asynchronous systems. Syst. ozbr. ta viysk. tehn. Vol. 27, 3, 60–63 (2011)
14. Perepelytsya, I., Zholtkevych, G.: Hierarchic Decomposition of Pre-machines as Models of Software System Components. Syst. upravl. navig. i zv'iazku. Vol. 20, 4, 233–238 (2011)

Models of Class Specification Intersection of Object-Oriented Programming

Dmitriy Buy¹ and Serhiy Kompan¹

Taras Shevchenko National University of Kyiv, Faculty of Cybernetics,
03680 Academician Glushkov Avenue 4d, Kyiv, Ukraine

buy@unicyb.kiev.ua, skompan@mail.ru

Abstract. This paper describes the application of heterogeneous algebraic system for the construction of the formal model of object database instead of object algebra. Complete formalization of the operation of intersection of class specifications is given.

Keywords. object-oriented programming, object database, object algebra, class specification

Key terms. MathematicalModel

1 Introduction

In applications of information technologies there is a problem of construction of the so-called dependable and stable systems and infrastructures – the systems which behave stably under all, especially, critical working circumstances. Similarity of risks and increasing actuality of their decline to an acceptable level for critical applications led to the appearance of a special term “safeware”, by the analogy with the terms “hardware”, “software”, “firmware” etc., which combines two components: *safe* – secure and *ware* – a product, an item. This term was suggested and patented by the leading expert of NASA on the questions of infrastructure security, professor N. Leveson, who registered the appearance of a modern field of knowledge called safeware engineering [1]. We mention a fundamental statement both obvious, and elusive in its nature: it’s impossible to talk about stability of a working system, especially of the infrastructure, if there is no formal model of its operation which has been constructed and verified. Moreover, for the construction of a formal model, more or less complex, not “toylike”, there should exist a mathematical apparatus with the help of which software developers create a formal model and verify it according to the source demands of a customer could.

For the full confidence in the fact that informational system will work stably (will be dependable and stable), one should single out system components, describe them formally and verify. Indeed, nowadays there is nothing instead of a “divide and rule”

approach to cope with this difficulty. In fact, one of the most important components of any complex system (infrastructure) is databases. That's why there should exist an appropriate formal model. For the relational databases such a formal model has been already constructed and explored considerably. This issue is exhaustively covered in the literature, beginning from the pioneering works by E.F. Codd (see, e.g. [2], the first textbooks [3, 4] and modern textbooks [5, 6]). We mention only a collection of works done by the collaborators of Taras Shevchenko National University of Kiev on the natural generalization of classical results of the databases relational approaches [7-14].

Nowadays, there are a lot of formal models of object-oriented databases (OODB) [15-20]. Each of these models elaborates OODB to a certain extent by applying certain mathematical apparatus. The analysis of research papers dedicated to OODB has shown that authors overlook the question arising from the necessity to construct a new class specification with the two given specifications. For example, the construction of a super class from two specified classes (the operation of intersection of class specification), the construction of a subclass from two super classes (the operation of union of class specification). The intersection of class specifications is important, in our opinion, as it provides for the opportunity to construct the core of a new program with two programs which allows integrating these two programs that results in the Framework version. This paper is dedicated to the exploration of the operation intersection of class specifications and refining conditions under which the intersection of classes is possible.

2 Practical results

The authors of this paper have conducted a number of investigations in the field under research: for example, in the article [21] it has been suggested to consider an object algebraic system as a model. Formally it can be formulated like this: $\langle O, K; \Omega_{obj}; \Omega_{spec}, \leq \rangle$, where O is a set of objects' classes, K is a set of class specification, Ω_{obj} is a set of operations over objects, Ω_{spec} is a set of operations over class specifications, and a relation $\leq \subseteq K \times K$ is a partial order which formalizes inheritance. The main objective of this article is specification of the intersection operation \cap and the difference of class specifications.

Let's start with the intersection operation \cap . Let us formalize the notion of a class: by a class we mean a pair $K = \langle s, \mu \rangle$, where s is a functional binary relation which associates an attribute with its meaning (from a universal domain D), and μ is a functional binary relation, which brings to conformity a method with its signature. Therefore [21], the relations s and μ determine a class specification.

The intersection operation (of class specifications) is an operation of the form $\cap: K \times K \rightarrow K$, where: $\langle s_1, \mu_1 \rangle \cap \langle s_2, \mu_2 \rangle = \langle s_1 \cap s_2, \mu_1 \cap \mu_2 \rangle$, where \cap is a standard set-theoretical intersection.

We will demonstrate some results concerning the structure of a partially ordered set (poset) $\langle F, \subseteq \rangle$, where F is a set of all the functional binary relations (on the universal domain X), a \subseteq is an ordinary set-theoretical inclusion. These results will supplement the results of the paper [22]. All undetermined notions and designations are understood in terms of this paper.

Lemma 1. For the arbitrary functional binary relations f and g the following equality is true: $f \cap g = (f \cap g) \upharpoonright (domf \cap domg)$ \square

Proof. ■ Let us start with $X \stackrel{def}{=} domf \cap domg$. Let us use generally valid properties of the set-theoretical restriction operation (a binary ratio on a set) (monotony, distributivity etc.) [19].

Firstly, we have an inclusion $dom(f \cap g) \subseteq domf \cap domg = X$. Secondly, from this the next chain of equalities and inequalities follows:

$$f \cap g = (f \cap g) \upharpoonright dom(f \cap g) \subseteq (f \cap g) \upharpoonright X = f \upharpoonright X \cap g \upharpoonright X \subseteq f \cap g.$$

$$\text{Thus, } f \cap g = (f \cap g) \upharpoonright X = (f \cap g) \upharpoonright (domf \cap domg) \square$$

Below \approx is a relation of consistency: $f \approx g \Leftrightarrow f \upharpoonright X = g \upharpoonright X$, where

$X \stackrel{def}{=} domf \cap domg$. In [7] the main property of consistency was determined as:

$$f \approx g \Leftrightarrow f \cup g \text{ is a functional binary relation.}$$

The following lemma's corollary forms another criterion of consistency.

Corollary (the criterion of consistency of functional binary relations). Let f, g be arbitrary functional binary relations, and $X \stackrel{def}{=} domf \cap domg$. Then: $f \approx g \Leftrightarrow dom(f \cap g) = X$, $\neg(f \approx g) \Leftrightarrow dom(f \cap g) \subset X$. \square

Proof. ■ The proof is performed by using a Lemma 1 and inclusion $dom(f \cap g) \subseteq X$. It's important to note that the second (the first) equivalence is a formal corollary of the first one (of the second one accordingly). \square

As for the structure of the poset $\langle F, \subseteq \rangle$, there are two statements.

Statement 1. Poset $\langle F, \subseteq \rangle$ is a lower semilattice, and at the same time, $\inf\{f, g\} = f \cap g$. \square

The proof results from the fact that F is a commutative idempotent semigroup and from a well-known connection between such semigroups and lower semilattices (see, e.g. [23]).

More complete information about the poset $\langle F, \subseteq \rangle$ is given by the following statement.

Statement 2. (the structure of poset $\langle F, \subseteq \rangle$). The following statements are true:

1. The empty function f_\emptyset is the smallest element ("a bottom")

2. The largest element in poset $\langle F, \subseteq \rangle$ exists if and only if the universe D is singleton
3. The infimum exists for any nonempty set F and $\inf F = \bigcap_{f \in F} f$
4. The supremum of the set F exists if and only if in the case when the set F is restricted, and $\sup F = \bigcup_{f \in F} f$
5. The element f is an atom only when f is singleton
6. Poset $\langle F, \subseteq \rangle$ is a relatively complete poset and a complete (upper) semilattice \square

Let's proceed to the substantial interpretation of above results.

The operation \cap constructs a new class which will be basic (paternal) for classes arguments. This intersection can also be empty, in this case we will get a special empty class.

As the relation \leq on the specifications is component wise

$(\langle s, \mu \rangle \leq \langle s', \mu' \rangle \Leftrightarrow s \subseteq s' \wedge \mu \subseteq \mu')$, all properties of the relation \subseteq (statements 1, 2) can be lifted to the relation \leq . The corresponding formulations are obvious and thereby are omitted.

3 Results and conclusions

The model of intersection operation of class specifications has been examined. This operation has been specified as set-theoretical intersection. The specification $f \cap g$ has been interpreted as the largest total part of f and g , that is, the specification from which specifications-arguments can be obtained by inheritance (in other words, the result specification is the specification of a paternal class). The conditions for nonempty (equivalent, empty) intersection have been examined.

As for formal results, natural criteria of function consistency have been presented (corollary) which supplement the already known criteria; the structure of a partially ordered set of partial functions has been specified (statements 1, 2).

References

1. Kharchenko, V. S.: Safety of Critical Infrastructures: Mathematical and Engineering Methods of Analysis and Ensuring. N.E. Zhukovsky National Aerospace University (2011) (in Russian)
2. Codd, E. F.: A Relational Model of Data for Large Shared Data Banks. Comm. ACM, 13 (1970)
3. Maier, D.: The Theory of Relational Databases. Computer Science Press (1983)
4. Ullman, J., Garsia-Molina, H., Widom, J.: Database Systems: The Complete Book. Prentice Hall Inc., Stanford (2002)
5. Kroenke, D. M.: Database Processing: Fundamentals, Design, and Implementation. Prentice Hall (2011)
6. Date, C. J.: An Introduction to Database Systems. In: Addison-Wesley, (2000)

7. Buy, D. B., Kahuta, N. D.: Full Image, Restriction, Projection, Relationship Compatibility. Theoretical and Applied Aspects of Program Systems Development: International Conference, December 8-10, pp. 244-260 (2009) (in Ukrainian)
8. Buy, D. B., Bogatiryova, J. A.: The Theory of Multisets: Bibliography, Use the Table in Databases. Radio Electronic and Computer Systems, 7(48), 56–62 (2010) (in Ukrainian)
9. Buy, D. B., Polyakov, S. A.: Compositional Semantics of Recursive Queries in SQL-like Languages. Bulletin of Kyiv University. Series. Phys.-Math. Science, 1, 45–56 (2010) (in Ukrainian)
10. Buy, D. B., Glushko, I. M.: Generalized Table Algebra, Generalized Tuple Calculus, Generalized Domain Calculus and Theirs Equivalence. In: Bulletin of Kyiv University. Series. Phys.-Math. Science. 1, 86–95 (2011) (in Ukrainian)
11. Buy, D. B., Puzikova, A. V.: Completeness of Armstrong Axioms. In: Bulletin of Kyiv University. Series. Phys.-Math. Science, 3, 103–108, (2011) (in Ukrainian)
12. Redko, V. N., Brona, J. Y., Buy, D. B., Polyakov, S. A.: Relational Databases: Tabular Algebra and SQL-like Language. AcademPeriodika (2001) (in Ukrainian)
13. Buy, D., Silveystruk, L.: Formalization of Structural Constraints of Relationships in «Entity-Relationship» Model. In: Electronic Computers and Informatics 2006: International Scientific Conference, September 20-22, pp. 96-101, Kosice, Slovakia (2006)
14. Buy, D., Glushko, I.: Equivalence of Table Algebras of Finite (Infinite) Tables and Corresponding Relational Calculi. In: Proceedings of the Eleventh International Conference on Informatics INFORMATICS'2011, November 16-18, pp. 56-60. Rožňava, Slovakia, (2011)
15. Piskunov, A. G.: The Formalization of the Object-Oriented Programming Paradigm, <http://www.realcoding.net/dn/docs/machine.pdf> (in Russian)
16. Piskunov, A. G.: The Formalization of the OOP: Types, Sets, Classes, <http://agp1.hx0.ru/articles/typeSetsClasses.pdf> (in Russian)
17. Chaplanova, E. B.: Operating Specification of Object-Relational Data Model. Radioelektronika, Informatika, Upravlinnya, 12, 75–79 (2011) (in Russian)
18. Richta, K., Toth, D.: Formal Models of Object-Oriented Databases. In: Objekty 2008. Žilina: Žilinská univerzita v Žiline, Fakulta Riadenia a Informatiky, pp. 204-217, <http://www.ksi.mff.cuni.cz/~richta/publications/richta-toth-Objekty2008.pdf> (2008)
19. Sarkar, M., Reiss, S.: A Data Model and a Query Language for Object-Oriented Database. In: Island, Department of Computer Science Brown University Providence, Rhode, CS-92-57, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.34.4531&rep=rep1&type=pdf> (1992)
20. Gail, M., Shaw, S.: Zdonik A Query Algebra for Object-Oriented Databases. Island, Department of Computer Science Brown University Providence, Rhode, CS-89-19 <http://trac.common-lisp.net/elephant/raw-attachment/wiki/RelationalAlgebra/shaw89query.2.pdf> (1989)
21. Buy, D. B., Kompan, S. V.: Union and Intersection Operations of Classes Specifications in Heterogen Algebraic System for Object-Oriented Programming. In: Proc. SWorld. Int. Sci-Pract. Conf. Modern Problems and Solutions in Science, Transportation, Manufacturing and Education. KUPRIENKO, Odessa, vol. 4, pp. 45–49 (2012) (in Russian)
22. Buy, D. B., Kahuta, N. D.: Properties Related Confinality and Order a Set of Partial Functions. Bulletin of Kyiv University. Series. Phys.-Math. Science, 2, 125–135, (2006) (in Ukrainian)
23. Skornyakov, L. A.: Elements of the Theory of Structures. Nauka, Moskow (1982) (in Russian)

Author Index

A

Alferov, Eugene.....	108
Alobaidi, Mizal.....	18
Arkatov, Denis B.....	178
Aronov, Andrey.....	252

B

Baiev, Oleksandr.....	118
Baklanova, Nadezhda.....	550
Batyiv, Andriy.....	18
Becker, Karsten.....	424
Beletsky, Aleksander.....	352
Beletsky, Anatoly.....	311, 352
Beletsky, Evgeny.....	311
Bilousova, Lyudmyla.....	209
Blinov, Igor Ol.....	565
Bodnenko, Dmitry.....	281
Bonda, Darya.....	360
Buy, Dmitriy.....	590

C

Chaabani, Mohamed.....	521
Cochez, Michael.....	221

D

Davidovsky, Maxim.....	99, 295
Derevianko, Andrii.....	30
Didenko, Ievgen.....	118
Doroshenko, Anatoliy.....	38
Dzyubenko, Artem.....	252

E

Echahed, Rachid.....	521
Ermolayev, Vadim.....	II, 64, 99, 108, 295

G

Glazunova, Olena G.....	411
Glukhovtsova, Kateryna.....	48
Guba, Anton.....	490

I

Isomöttönen, Ville.....	221
Itkonen, Jonne.....	221
Iurtyn, Ivan.....	187
Ivanov, Ievgen.....	448

K

Kandyba, Roman.....	352
Keberle, Natalya G.....	79
Kharchenko, Vyacheslav.....	146
Klionov, Dmitriy M.....	464
Kobets, Vitaliy.....	II, 310, 329
Kolgatin, Oleksandr.....	209
Kolgatina, Larisa.....	209
Kompan, Serhiy.....	590
Kotkova, Vera.....	236
Kravtsov, Hennadiy.....	II, 236, 410
Kropotov, Aleksandr.....	30
Kryukov, Sergey.....	310
Kryvolap, Andrii.....	533
Kukharenko, Vladimir.....	273, 410

Kuliamin, Victor 573
 Kushnir, Nataliya 195
 Kuzminska, Olena 264

L

Lavrisheva, Ekaterina 252
 Lazareva, Elena 339
 Lazurik, Valentine 118
 Letichevsky, Alexander 4

M

Maksimov, Andrey 573
 Mallet, Frédéric 130, 289, 475
 Mantula, Elena 91
 Manzhulam Anna 195
 Mashtalir, Vladimir 91
 Matthes, Ralph 506
 Matzke, Wolf-Ekkehard 2
 Mayr, Heinrich C. II
 Mazol, Sergey 360
 Mazol, Sergey 366
 Mesropyan, Karine 385
 Mikhailova, Irina 582
 Moiseeva, Oksana 366
 Möller, Dietmar P.F. 424
 Morze, Natalia 264
 Morze, Natalia V. 411

N

Nikitchenko, Mykola II, 447, 533
 Novikov, Boris 582

O

Odarushchenko, Oleg 146
 Odarushchenko, Valentina 146

P

Payentko, Tanya 310
 Peschanenko, Vladimir II, 447, 490
 Petrenko, Alexander K. 573
 Petukhova, Lyubov 236

Popov, Peter 146
 Pratt, Gary L. 3
 Protsenko, Galina 264

R

Ralo, Aleksandr 30
 Richter, Harald 424
 Romenska, Yuliia 130
 Rudenko, Margarita 401

S

Schreiner, Wolfgang 533
 Selyutin, Victor 401
 Semenyuk, Andriy 393
 Shushpanov, Constantin 490
 Shyshkina, Mariya 436
 Sitzmann, Daniel 424
 Sokol, Vladyslav 48
 Spivakovska, Evgeniya 236
 Spivakovskiy, Aleksander II, 236
 Strecker, Martin 447, 521, 550
 Styervoyedov, Sergiy 30

T

Tatarintseva, Olga 64
 Tirronen, Ville 221
 Tkachuk, Nikolay 48
 Tolok, Vyacheslav 99

V

Valko, Nataliya 195
 Varava, Anastasiia 163
 Vasylevych, Leonid 187
 Vozniy, Oleksiy 30

W

Weissblut, Alexander J. 374
 Winckel, Mathias 506

Y

Yatsenko, Olena 38

Z

Zaporozhchenko, Yulia.....	410	Zhereb, Kostiantyn.....	38
Zaretska, Iryna.....	475	Zholtkevych, Galyna.....	475
Zavileysky, Mikhail.....	II	Zholtkevych, Grygoriy.....	II, 18, 163, 475, 582