Steffen Hölldobler
Yanchun Liang
(Eds.)

1st International Workshop on Semantic Technologies (IWOST)

Proceedings

March 9–12 2015

Jilin University, Changchun, China

Volume Editors

Steffen Hölldobler
International Center for Computational Logic
Technische Universität Dresden
01062 Dresden, Germany
email: sh@iccl.tu-dresden.de

Yanchun Liang
Jilin University
Changchun, China
Email: ycliang@jlu.edu.cn

Preface

This volume contains the papers presented at the *First International Workshop on Semantic Technologies (IWOST)* held on March 09-12, 2015 at the Jilin University in Changchun, China. The workshop is an activity of the cluster *Semantic Technologies* with the project *Swap and Transfer.* This project is funded by the EU within the Erasmus Mundus programme and aims at fostering sustainable development, innovation and technology transfer.

There were 14 submissions. Each submission was reviewed by at least 2 program committee members. The committee decided to accept 11 papers. The program also includes invited talks by Fausto Guinchiglia from the University of Trento on *Open Data Integration, Cleaning and Reuse* and by Chagnaa Altangerel from the National University of Mongolia on *Language Resources in the Semantic Web.*

The editors would like to thank Peter Steinke for his help in setting up the web page and compiling this proceedings as well as the local organizers at Jilin University without which the workshop would not have been possible. We thank the *Swap and Transfer* project as well as Jilin University for supporting the workshop financially. Finally, we are indepted to Easy Chair which made the organization of the workshop easy for the chairs.

March 6, 2015
Dresden and Changchun

Steffen Hölldobler
Yanchun Liang

Program Committee

Idilia Bachkova	Chimikotehnologitchen I Metalurgitchen Universitet
Khang Tran Dinh	Hanoi University of Science and Technology
Anna Fensel	University of Innsbruck
Steffen Hölldobler	Technische Universitaet Dresden
Yanchun Liang	University of Jilin
Maurizio Marchese	University of Trento
Thai Ha Phi	Hanoi University of Transport and Communication
Srun Sovila	Royal University of Phnom Phen
Hao Xu	University of Jilin

Additional Reviewers

Norbert Manthey
Tobias Philipp

Contents

Exploring Trends of Cancer Research Based on Topic Model

Mingliang Cui, Yanchun Liang*, Yuping Li, Renchu Guan*
College of Computer Science and Technology, Jilin University,
Changchun, 130012, China
{Yanchun Liang ycliang@jlu.edu.cn; Renchu Guan guanrenchu@jlu.edu.cn}

Abstract. Cancer research is of great importance in life science and medicine and attracts research funds of thousands of millions dollars each year. With the explosion of biomedical research papers, it becomes more and more necessary to show the research trend in this spotlight area. In this paper, to provide a straightforward research atlas for the top killer cancers, Latent Dirichlet Allocation (LDA) is performed on the massive quantities of biomedical literatures. Moreover, Gibbs Sampling is used to make assessment on the parameters of the LDA model. The proposed evaluation carried out under multiple conditions with different Ks (the number of topics) for the top five cancers in recent five years. Additionally, a biomedical topic model was generated with the LDA model and delicate analysis was performed on the basis of that in order to explore the trending topic in cancer research. It can help the biology and medicine doctors quickly catch the frontiers of the cancer study, improve and expand their research programme, especially in today's era of "big data".

Key words: Topic Model, LDA, Gibbs Sampling, Topic Analysis, Cancer Research Trend

1 Introduction

1.1 Significance

Cancer is of great threat to human health, scientists and Medicians have been continuously looking for effective treatment to conquer it, however, cancer research is a very challenging field in today's life science. Over the past 5 years cancer research has diverged enormously, partly based on the quickly development of biotechnology and bioinformatics. It is not easy to summarize recent trends for different cancers' study, and identify where the new findings are and therapies might come from [1]. Under this circumstance, we choose an appropriate machine learning method—topic model—to explore the trend in cancer research, specifically, the topics in cancer research papers.

The topic model is a probabilistic model of text mining appeared in recent years [2]. It is an algorithm that can discover the topic structure hidden in large-scale data. In topic model, the vocabulary items is visible while the topic structure hidden. In order to reduce the dimensions of the feature vector space, texts are usually mapped into the topic space via the topic model. It is different from the traditional vector space model, which just simply considers each document as a sample and each word as a feature. Instead, it maps a high dimensional frequency space to a lower dimensional topic space. Moreover, the topic model can capture the semantic information, which can reveal that the latent relations among documents. It also can effectively solve the polysemy, synonym and other problems, which has the vital significance in document feature extraction and content analysis. However, using a topic model (i.e. Latent Dirichlet Allocation) to analyze the trends of cancers has not been reported.

The rest of the article is organized as follows: we start in section 2 with a brief review of a topic model named as Latent Dirichlet Allocation and its related work. In section 3, the algorithm of Gibbs Sampling is introduced, and the framework of our exploration is described in detail. It presents the experimental methodology and results on Medline dataset in Section 4. At last, conclusions and future work are depicted in Section 5.

2 Background

In 1998, Professor Papadimitriou proposes LSI (Latent Semantic Indexing) model [3], which can be seen as the origin of the Topic model. In 1999, Professor Hofmann put it forward to pLSI (Probabilistic Latent Semantic Indexing) model [4] and LDA (Latent Dirichlet Allocation) is a generalization of pLSI, which adds Dirichlet Bias generating prior distribution model, proposed by Blei in 2003[5].

LDA has been widely applied in information retrieval, text mining, Natural Language Processing fields and has become a research hotspots recently [6-8]. In this paper, LDA model is introduced to analyze the Medline data especially on cancer research. In the LDA model, a document is generated as follows [9]:

1. First, generate the topic distribution of the document *i* by sampling from the dirichlet distribution α.

2. Second, generate the topic of the word *j* in the document *i* by sampling from the topic distribution.

3. Then generate the word distribution of the topic by sampling from the Dirichlet distribution β.

4. Finally generate the word *j* in the document *i* by sampling from the word distribution.

Which is similar to the binomial distribution Beta distribution [10] is the conjugate prior probability distribution [11], while the Dirichlet distribution is a polynomial distributed conjugate prior probability distribution.

The structure diagram of LDA model is shown in the Figure.1 (similar to the Bayesian network structure [12]):

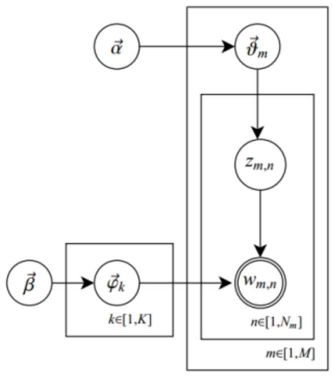


Fig. 1. LDA model structure diagram which visualize the generation process using the LDA

Hyper parameters are subject to Dirichlet distribution. We explain the symbol used to describe the model in Table.1. Now we have a text representation of the probability:

$$p(\vec{w}_m, \vec{z}_m, \vec{\theta}_m, \Phi \mid \vec{\alpha}, \vec{\beta}) = \prod_{n=1}^{N_m} p(w_{m,n} \mid \vec{\varphi}_{z_{m,n}}) p(z_{m,n} \mid \vec{\theta}_m) \cdot p(\vec{\theta}_m \mid \vec{\alpha}) \cdot p(\Phi \mid \vec{\beta}) \quad \textbf{(1)}$$

Table 1. A table of symbols used to describe the LDA model.

Symbol	Meaning
M	document collection
K	topic collection
V	term collection
N_m	length of the document *m*
α	dirichlet prior of the topic distribution, hyper parameter
β	dirichlet prior of the term distribution, hyper parameter
α_i	*i* component of α
β_i	*i* component of β
θ	topic distribution generated by $Dir(\alpha)$
θ_m	document m distribution $p(z\|d = m)$
φ_m	term distribution generated by $Dir(\beta)$
φ_k	term distribution of topic *k* $p(t\|z = k)$
z	topic number mapping to φ_z in topic collection K
$w_{m,n}$	word *n* of the document *m*
$z_{w_{m,n}}$	topic of word *n* of the document *m*
Φ	generated topic collection $\Phi = \{\varphi_k\}_{k=1}^{K}$
n_m	topic vector of document *m*
n_k	term vector of topic *k*
n_k^t	*t* component of n_k
n_m^k	*k* component of n_m
$n_{m,\neg i}^k$	ditto
$n_{k,\neg i}^t$	ditto
$\hat{L}$	the evaluation parameter of the model

The convergence of LDA is a more critical issue. We use the convergence function to solve it. Under the given model conditions, we choose the appearance probability of samples as the evaluation criterion of the model. The performance of the LDA model:

$$p(\vec{\mathrm{w}} \mid \mathrm{M}) = \prod_{m=1}^{M} p(\vec{\mathrm{w}}_m \mid \vec{\theta}_m, \Phi) \tag{2}$$

$$= \prod_{m=1}^{M} \prod_{n=1}^{N_m} \sum_{k=1}^{K} \left(\frac{n_k^t + \beta_t}{\sum_{t=1}^{V} n_k^t + \beta_t} \cdot \frac{n_m^k + \alpha_k}{\sum_{k=1}^{K} n_m^k + \alpha_k} \right). \tag{3}$$

For the convenience of the calculation, we use log transform to the equation (1), denoted as $\hat{L}$. Along with the iterative constantly, $\hat{L}$ is used to determine the model convergence [13].

$$\hat{L} = -\sum_{m=1}^{M} \sum_{n=1}^{N_m} \log_2 \left(\sum_{k=1}^{K} \left(\frac{n_k^t + \beta_t}{\sum_{t=1}^{V} n_k^t + \beta_t} \cdot \frac{n_m^k + \alpha_k}{\sum_{k=1}^{K} n_m^k + \alpha_k} \right) \right) \tag{4}$$

3 Main Work

3.1 Core Method

In fact, LDA is one of the probabilistic. Data are generally divided into two parts, visible variables and latent variables. It is believed in the topic model that the data is produced by the generation process, which defines the joint probability distribution of visible random variables and latent random variables. For many modern probabilistic models, including Bayesian statistics, the priori probability calculation is extremely difficult. So the core research objective of modern probability modelling is to do everything possible to obtain an approximate solution. Random sampling is a kind of methods for solving the approximate solution with good performance. This article describes a method commonly used sampling MCMC (Markov Chain Monte Carlo) [14] and Gibbs Sampling algorithm [15], Gibbs Sampling algorithm has been widely used in modern Bayesian analysis.

MCMC methods have Gibbs Sampling algorithm and Metropolis-Hastings (MH) [16] algorithm commonly used sampling methods. Gibbs Sampling algorithm is a special case of MH algorithm, Gibbs Sampling from a high-dimensional space are sampled separately for each dimension, and gradually get higher dimensional sampling points, making sampling difficult to reduce.

N-dimensional Gibbs Sampling:

1. Random initialization $\{ x_i : i = 1, 2, \cdots, n \}$

2. For $t = 0, 1, 2, \cdots$ loop in sampling

$x_1^{(t+1)} \sim p(\mathrm{x}_1 \mid \mathrm{x}_2^t, \mathrm{x}_3^t, \cdots, \mathrm{x}_n^t)$

$$\cdots$$

$$X_j^{(t+1)} \sim p(\mathrm{x}_j \mid \mathrm{x}_1^{(t+1)}, \cdots, \mathrm{x}_{j-1}^{(t+1)}, \mathrm{x}_{j+1}^{t}, \cdots, \mathrm{x}_n^{t})$$

$$\cdots$$

$$X_n^{(t+1)} \sim p(x_n \mid x_1^{t+1}, x_2^{t+1}, \cdots, x_{n-1}^{t+1})$$

3.2 Complete Process

We choose the top 5 cancer from the top 10 deadliest cancer published by the LiveScience [17], which are Breast Cancer, Lung Cancer, Pancreatic Cancer, Prostate Cancer and Colon Cancer. We search and download the research paper related to these cancers from NCBI PubMed from 2010 to 2014 separately [18].

Then we make a pre-process to extract the title and abstract for each paper and make some hyphens for the entity name for a better segmentation. After that we use WordNet [19] to stem the text in order to get an exact description for the topic. Before the document input into model, we will do the pre-processing for the document, thereby obtaining the document term matrix. It can be seen by term matrix that how many documents in corpus, how many word terms and how frequently each term appears in a document. The inputs of the model are the document collection M, the number of topics K, and the hyper parameters α, β.The topic of a number K we need to specify its value according to the experience, we want to take advantages of K solution, the need for repeated experiments, and then to carry on the value according to the different K value under the situation of convergence. After repeated experiments of LDA convergence on the data, we get a reasonable value of K=100. The hyper parameter α, β is the Dirichlet of the prior distribution. In fact they have smooth effect on data. Because there is no supervision information too much, we assign the hyper parameters an empirical value $\alpha_k = 0.5, \beta_t = 0.1$, tending to take symmetry value [20].

Take the dataset of Pancreatic Cancer in 2010 as an example, the number of the abstract is 2088, includes 54004 words. The number of topic K is set to 100, 200, 300, 400, 500, considered for model convergence condition.

According to the evaluation function of the convergence $\hat{L}$ mentioned in Section 2, which shows the cost of compression of the text using the model, the smaller the better. Figure.2 below respectively under different K values for the iterative convergence condition of iteration times of 400 and 1000.

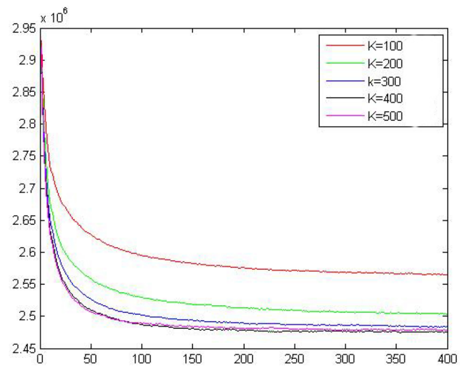


Fig. 2. The 400 iteration convergence condition of the LDA Gibbs Sampling model. X-axis is iterations, Y-axis is results of convergence function.

We further analysis the output of the LDA Gibbs Sampling, particularly in the topic words file (this file contains topic words most likely words of each topic). We first transform the topic-word matrix to a topic word vector representing the selection and the frequency of topic words to describe one cancer research. Then we use feature scaling normalization [21] (formula 5) to deal with the word vector, in order to compare them between different years and different cancers.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad \textbf{(5)}$$

We measure the similarity and differences between the 5 cancer research by their word vector cosine coefficients similarity [22] (formula 6).

$$\text{simlarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} Ai \times Bi}{\sqrt{\sum_{i=1}^{n}(Ai)^2} \times \sqrt{\sum_{i=1}^{n}(Bi)^2}} \qquad \textbf{(6)}$$

Unlike the cosine coefficients which give a numerical description on the trend and topic of the 5 cancers, the common words are easier for readers to have an intuition on the 5 cancers topics. We do the both analysis of the results so that we may get a comprehensive answer.

4 Experiments and Discussion

To examine the behavior and the performance of LDA, the experiments are illustrated on the widely used Medline dataset. In order to straightforwardly compare the trend and topic resulted from the LDA model, the visual results to get the entire recognition of the topic words of different cancers are shown, which uses the Word Cloud tool supported by Tagul.com [24]. And we also draw the trend of cancer research in different years, supported by Plot.ly [25].

4.1 Experimental Setup

The publicly available Medline dataset is provided by NCBI PubMed, which contains a title and an abstract for each paper on the five fatal cancers. In order to get a better visual understanding of the data, the 5 cancers with different colors (in details, breast cancer blue, colon cancer orange, lung cancer green, pancreatic cancer red, prostate cancer purple) are listed in Table 2. An outline of the selected dataset is shown below (actually we collect the 2014 data before October 2014, that is why it seems all the 2014 paper decrease from the previous years):

Table 2. A table reports the statistics of papers on the 5 cancer published from 2010 to 2014

Cancer	2010	2011	2012	2013	2014
Breast	11558	12810	13908	**14619**	11421
Colon	4484	4852	5386	**5498**	4305
Lung	7609	8530	9615	**10572**	8896
Pancreas	2088	2367	2667	**2860**	2468
Prostate	6365	7165	7759	**8137**	6787

4.2 Experiment Results

After introducing the Gibb Sampling and transforming the topic-word matrix to the vector, we draw all the five cancer topic words in the recent five years. A snapshot of the collection of the topic words clouds are shown in Figure.3.

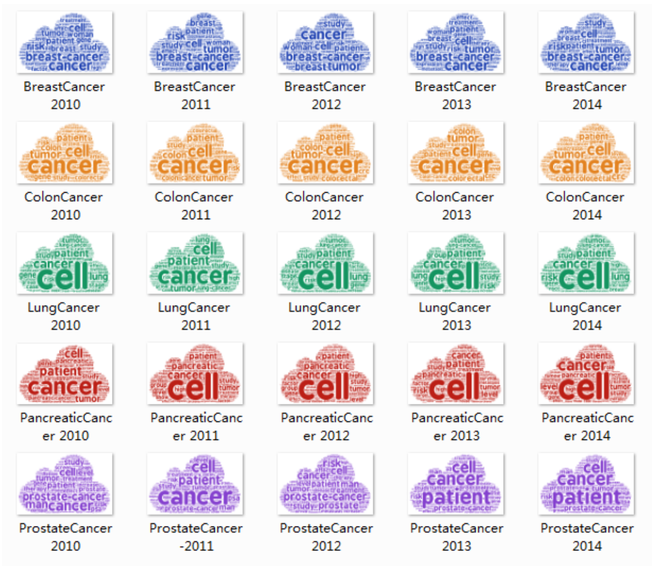


Fig. 3. A snapshot about the topic words for each cancer research for the 5 years. Each row is a type of cancer, and each column is a certain year. The 5 cancers are drawn with different colors. Also, the more frequently the word appears, the more striking the word displays.

Filtering the topic words for each cancer into the common words collection, we can easily discover the main topic for each cancer and new findings and focus for each year. Take breast cancer as an example shown below:

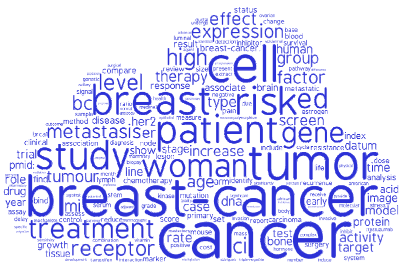


Fig. 4. The word cloud of the common topic words for breast cancer research during 2010-2014.

In the Figure.4 above, we can find out the big words in it like cell, breast cancer, tumor, patient, woman and so on, these words are simply the most common used to elaborate breast cancer, which means if you talk about breast cancer and you just can’t avoid mentioning them. They are obviously from the different topics, therapy, factor, risk and other aspects of the breast cancer. The common words are defined as the words which appear every year for certain cancer research during 2010-2014. They are shown in Figure.5 below, and the shape of the words represent their frequency.

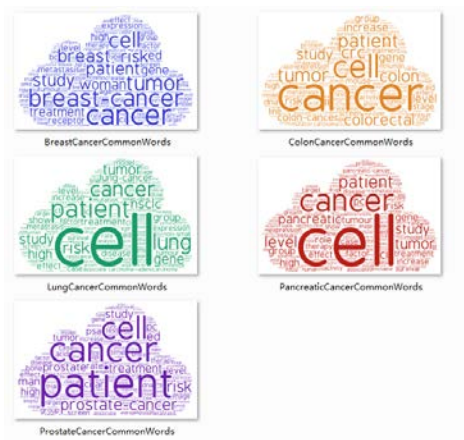


Fig. 5 The word cloud of the common topic words for the top 5 deadliest cancer research during 2010-2014.

4.3 General Comparison and Discussions

The Figure.6 above shows the contrast result of the vector cosine coefficient between two cancers in the same year. The higher coefficient, the more similar. We can easily find out that all the similarities contrast with colon cancer (see colon&lung, colon & pancreatic, breast&colon, colon&prostate) are higher than the other cancer pairs. It indicates colon cancer research is more related with the others. It is because that: lung cancer may easily metastasize to colon; colon cancer and pancreatic cancer are both belong to lower digestive cancers; lack of exercise and sitting for long time are the common causes of breast and colon cancer; long-term androgen deprivation therapy for prostate cancer may increase the risk of colon cancer.

And we find it interesting that almost all the five cancer research diverse from each other in 2012, because they have a concave at the time in the figure. Fig.7 shows the topic words changes for each cancer research during the recent five years. We can learn from the figure that breast cancer, lung cancer, prostate cancer, these 3 cancer research only change a little, and pancreatic cancer research changes a lot. It is because that the pancreatic incidence rate increase fast recently and it is named as "the king of cancer" because the lowest survival rate. Many scientists engage in the research of pancreatic cancer.

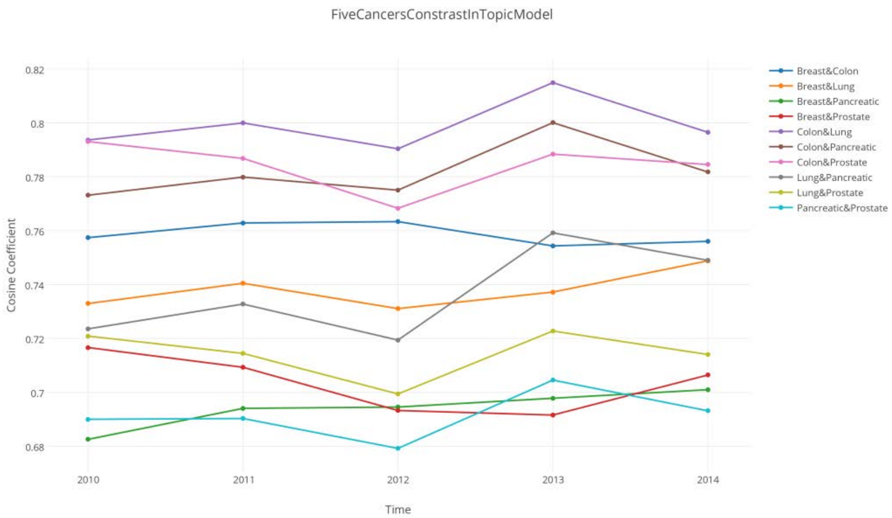


Fig. 6. All five cancers intercomparsion every year during 2010-2014. X-axis is year time, Y-axis is cosine coefficient.

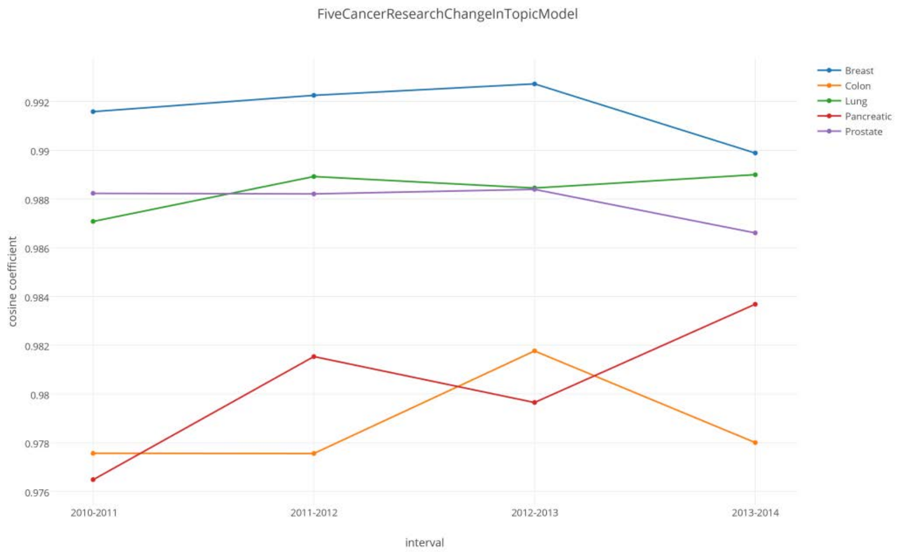


Fig.7. All five cancers change by calculating the topic vector similarity with their former year. X-axis is interval, Y-axis is cosine coefficient.

5 Conclusions

In this paper, we first apply LDA Gibbs Sampling model on the analysis of the top 5 deadliest cancer research trends, which is extended from cosine coefficient using the vector space model after transforming topic-word matrix into topic word vector. Then we generate the common topic word collection for each cancer research, in order to get the trending topic words. We further explore the trend by comparing and contrasting topic words for each cancer and their cosine coefficients. It is found that the trending topic words for the 5 cancers research from 2010 to 2014, which are depicted in the words clouds. Moreover, it is found that numerical trends of the four cancers research are as follows: breast cancer, lung cancer, and prostate cancer are of little change, but pancreatic cancer is changing a lot in the recent five years. But what cause all five cancer research diverse in 2012, how to visualize the allocation of the topics for each cancer research, and how to make a computational evaluation for the trend results, are all what need to be explored in our future work.

Acknowledgments.
The authors are grateful to the support of the NSFC (61272207, 61472158, 61103092) and the Science Technology Development Project from Jilin Province (20130522106JH, 20140520070JH）

References

1. Cao, Y., DePinho, R., Ernst, M., Vousden, K.: Cancer research: past, present and future. Nature Reviews Cancer. 11, 749–754 (2011).
2. Rosen-Zvi, M., Griffiths, T., Steyvers, M., & Smyth, P.: The author-topic model for authors and documents. Proceedings of the 20th conference on Uncertainty in artificial intelligence. 487-494 (2004).
3. Papadimitriou, C. H., Tamaki, H., Raghavan, P., & Vempala, S.: Latent semantic indexing: A probabilistic analysis. Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems. 159-168 (1998).
4. Hofmann, T.: Probabilistic latent semantic indexing. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. 50-57 (1999).
5. Blei, D. M., Ng, A. Y., & Jordan, M. I.: Latent dirichlet allocation. the Journal of machine Learning research, 3, 993-1022 (2003).
6. Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M.: Hierarchical dirichlet processes. Journal of the american statistical association 101.476 (2006).
7. Mcauliffe, Jon D., David M. Blei.: Supervised topic models. Advances in neural information processing systems. 121-128 (2008).
8. Petinot, Yves, Kathleen McKeown, and Kapil Thadani.: A hierarchical model of web summaries. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. 670-675. Association for Computational Linguistics (2011).
9. Diebolt, Jean, Christian P. Robert.: Estimation of finite mixture distributions through Bayesian sampling. Journal of the Royal Statistical Society. Series B (Methodological). 363-375 (1994).
10. S. Kotz, N. Balakrishnan, N. L. Johnson: Dirichlet and Inverted Dirichlet Distributions. Continuous Multivariate Distributions, Models and Applications. New York: Wiley, 2000., United States (2004).
11. Lindley, Dennis V.: The use of prior probability distributions in statistical inference and decision. Proc. 4th Berkeley Symp. on Math. Stat. and Prob. 453-468 (1961).
12. Beal, Matthew James.: Variational algorithms for approximate Bayesian inference. PhD diss. University of London (2003).
13. Heinrich, Gregor.: Parameter estimation for text analysis. Technical report. (2005).
14. Gilks, Walter R.: Markov chain monte carlo. John Wiley & Sons, Ltd. (2005).
15. Porteous, Ian, David Newman, Alexander Ihler, Arthur Asuncion, Padhraic Smyth, Max Welling.: Fast collapsed gibbs sampling for latent dirichlet allocation. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 569-577 (2008).
16. Chib S., Greenberg E.: Understanding the metropolis-hastings algorithm. The american statistician. 49(4): 327-335 (1995).

17. Chan, A.: The 10 Deadliest Cancers and Why There's No Cure, http://www.livescience.com/11041-10-deadliest-cancers-cure.html.
18. Ncbi.nlm.nih.gov,: Home - PubMed - NCBI, http://www.ncbi.nlm.nih.gov/pubmed.
19. University, P.: About WordNet - WordNet - About WordNet, http://wordnet.princeton.edu/.
20. Griffiths, T., Steyvers: Finding scientific topics. Proceedings of the National Academy of Sciences. 101, 5228–5235 (2004).
21. Wikipedia,: Normalization, http://en.wikipedia.org/wiki/Normalization.
22. Wikipedia,: Cosine similarity, http://en.wikipedia.org/wiki/Cosine_similarity.
23. O'Rourke, Norm, R. Psych, Larry Hatcher.: A step-by-step approach to using SAS for factor analysis and structural equation modeling. Sas Institute. (2013).
24. Tagul.com,: Tagul - Gorgeous word clouds, http://tagul.com.
25. Plot.ly,: Plotly, https://plot.ly/.

On Indicative Conditionals

Emmanuelle Anna Dietz[1], Steffen Hölldobler[1], and Luís Moniz Pereira[2⋆]

[1]International Center for Computational Logic, TU Dresden, Germany,
{dietz,sh}@iccl.tu-dresden.de
[2]NOVA Laboratory for Computer Science and Informatics, Caparica, Portugal,
lmp@fct.unl.pt

Abstract. In this paper we present a new approach to evaluate indicative conditionals with respect to some background information specified by a logic program. Because the weak completion of a logic program admits a least model under the three-valued Łukasiewicz semantics and this semantics has been successfully applied to other human reasoning tasks, conditionals are evaluated under these least Ł-models. If such a model maps the condition of a conditional to *unknown*, then abduction and revision are applied in order to satisfy the condition. Different strategies in applying abduction and revision might lead to different evaluations of a given conditional. Based on these findings we outline an experiment to better understand how humans handle those cases.

1 Indicative Conditionals

Conditionals are statements of the form *if condition then consequence*. In the literature the condition is also called *if part*, *if clause* or *protasis*, whereas the consequence is called *then part*, *then clause* or *apodosis*. Conditions as well as consequences are assumed to be finite sets (or conjunctions) of ground literals.

Indicative conditionals are conditionals whose condition may or may not be true and, consequently, whose consequence also may or may not be true; however, the consequence is asserted to be true if the condition is true. Examples for indicative conditionals are the following:

If it is raining, then he is inside. (1)

If Kennedy is dead and Oswald did not shoot him, then someone else did. (2)

If rifleman A did not shoot, then the prisoner is alive. (3)

If the prisoner is alive, then the captain did not signal. (4)

If rifleman A shot, then rifleman B shot as well. (5)

If the captain gave no signal and rifleman A decides to shoot,
then the prisoner will die and rifleman B will not shoot. (6)

Conditionals may or may not be true in a given scenario. For example, if we are told that a particular person is living in a prison cell, then most people are

⋆ The authors are mentioned in alphabetical order.

expected to consider (1) to be true, whereas if we are told that he is living in the forest, then most people are expected to consider (1) to be false. Likewise, most people consider (2) to be true.

The question which we shall be discussing in this paper is how to automate reasoning such that conditionals are evaluated by an automated deduction system like humans do. This will be done in a context of logic programming (cf. [11]), abduction [9], Stenning and van Lambalgen's representation of conditionals as well as their semantic operator [19] and three-valued Łukasiewicz logic [12], which has been put together in [6,7,5,8,3] and has been applied to the suppression [2] and the selection task [1], as well as to model the belief-bias effect [15] and contextual abductive reasoning with side-effects [16].

The methodology of the approach presented in this paper differs significantly from methods and techniques applied in well-known approaches to evaluate (mostly subjunctive) conditionals like Ramsey's belief-retention approach [17], Lewis's maximal world-similarity one [10], Rescher's systematic reconstruction of the belief system using principles of saliency and prioritization [18], Ginsberg's possible worlds approach [4] and Pereira and Aparíco's improvements thereof by requiring relevancy [14]. Our approach is inspired by Pearl's do-calculus [13] in that it allows revisions to satisfy conditions whose truth-value is unknown and which cannot be explained by abduction, but which are amenable to hypothetical intervention instead.

2 Preliminaries

We assume the reader to be familiar with logic and logic programming. A *(logic) program* is a finite set of (program) clauses of the form $A \leftarrow B_1 \wedge \ldots \wedge B_n$ where A is an atom and B_i, $1 \leq i \leq n$, are literals or of the form $\top$ and $\bot$, denoting truth- and falsehood, respectively. A is called *head* and $B_1 \wedge \ldots \wedge B_n$ is called *body* of the clause. We restrict terms to be constants and variables only, i.e., we consider so-called data logic programs. Clauses of the form $A \leftarrow \top$ and $A \leftarrow \bot$ are called *positive* and *negative facts*, respectively.

In the this paper we assume for each program that the alphabet consists precisely of the symbols mentioned in the program. When writing sets of literals we will omit curly brackets if the set has only one element.

Let $\mathcal{P}$ be a program. $g\mathcal{P}$ denotes the set of all ground instances of clauses occurring in $\mathcal{P}$. A ground atom A is *defined* in $g\mathcal{P}$ iff $g\mathcal{P}$ contains a clause whose head is A; otherwise A is said to be *undefined*. Let $\mathcal{S}$ be a set of ground literals. $\mathit{def}(\mathcal{S}, \mathcal{P}) = \{A \leftarrow \mathit{body} \in g\mathcal{P} \mid A \in \mathcal{S} \vee \neg A \in \mathcal{S}\}$ is called *definition* of $\mathcal{S}$.

Let $\mathcal{P}$ be a program and consider the following transformation:

1. For each defined atom A, replace all clauses of the form $A \leftarrow \mathit{body}_1, \ldots, A \leftarrow \mathit{body}_m$ occurring in $g\mathcal{P}$ by $A \leftarrow \mathit{body}_1 \vee \ldots \vee \mathit{body}_m$.
2. If a ground atom A is undefined in $g\mathcal{P}$, then add $A \leftarrow \bot$ to the program.
3. Replace all occurrences of $\leftarrow$ by $\leftrightarrow$.

The ground program obtained by this transformation is called *completion* of $\mathcal{P}$, whereas the ground program obtained by applying only the steps 1. and 3. is called *weak completion* of $\mathcal{P}$ or $wc\mathcal{P}$.

We consider the three-valued Łukasiewicz (or Ł-) semantics [12] and represent each interpretation I by a pair $\langle I^\top, I^\bot\rangle$, where $I^\top$ contains all atoms which are mapped to *true* by I, $I^\bot$ contains all atoms which are mapped to *false* by I, and $I^\top \cap I^\bot = \emptyset$. Atoms occurring neither in $I^\top$ not in $I^\bot$ are mapped to *unknown*. Let $\langle I^\top, I^\bot\rangle$ and $\langle J^\top, J^\bot\rangle$ be two interpretations. We define

$$\langle I^\top, I^\bot\rangle \subseteq \langle J^\top, J^\bot\rangle \text{ iff } I^\top \subseteq J^\top \text{ and } I^\bot \subseteq J^\bot.$$

Under Ł-semantics we find $F \wedge \top \equiv F \vee \bot \equiv F$ for each formula F, where $\equiv$ denotes logical equivalence. Hence, occurrences of the symbols $\top$ and $\bot$ in the bodies of clauses can be restricted to those occurring in facts.

It has been shown in [6] that logic programs as well as their weak completions admit a least model under Ł-semantics. Moreover, the least Ł-model of the weak completion of $\mathcal{P}$ can be obtained as least fixed point of the following semantic operator, which was introduced in [19]: $\Phi_\mathcal{P}(\langle I^\top, I^\bot\rangle) = \langle J^\top, J^\bot\rangle$, where

$$\begin{aligned} J^\top &= \{A \mid A \leftarrow \mathit{body} \in g\mathcal{P} \text{ and } \mathit{body} \text{ is } \mathit{true} \text{ under } \langle I^\top, I^\bot\rangle\}, \\ J^\bot &= \{A \mid \mathit{def}(A,\mathcal{P}) \neq \emptyset \text{ and} \\ &\qquad \mathit{body} \text{ is } \mathit{false} \text{ under } \langle I^\top, I^\bot\rangle \text{ for all } A \leftarrow \mathit{body} \in \mathit{def}(A,\mathcal{P})\}. \end{aligned}$$

We define $\mathcal{P} \models_Ł^{lmwc} F$ iff formula F holds in the least Ł-model of $wc\mathcal{P}$.

As shown in [2], the Ł-semantics is related to the well-founded semantics as follows: Let $\mathcal{P}$ be a program which does not contain a positive loop and let $\mathcal{P}^+ = \mathcal{P} \setminus \{A \leftarrow \bot \mid A \leftarrow \bot \in \mathcal{P}\}$. Let u be a new nullary relation symbol not occurring in $\mathcal{P}$ and B be a ground atom in

$$\mathcal{P}^* = \mathcal{P}^+ \cup \{B \leftarrow u \mid \mathit{def}(B,\mathcal{P}) = \emptyset\} \cup \{u \leftarrow \neg u\}.$$

Then, the least Ł-model of $wc\mathcal{P}$ and the well-founded model for $\mathcal{P}^*$ coincide.

An *abductive framework* consists of a logic program $\mathcal{P}$, a set of *abducibles* $\mathcal{A}_\mathcal{P} = \{A \leftarrow \top \mid A \text{ is undefined in } g\mathcal{P}\} \cup \{A \leftarrow \bot \mid A \text{ is undefined in } g\mathcal{P}\}$, a set of *integrity constraints* $\mathcal{IC}$, i.e., expressions of the form $\bot \leftarrow B_1 \wedge \ldots \wedge B_n$, and the entailment relation $\models_Ł^{lmwc}$, and is denoted by $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_Ł^{lmwc}\rangle$.

One should observe that each finite set of positive and negative ground facts has an Ł-model. It can be obtained by mapping all heads occurring in this set to *true*. Thus, in the following definition, explanations are always satisfiable.

An *observation* $\mathcal{O}$ is a set of ground literals; it is *explainable* in the abductive framework $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_Ł^{lmwc}\rangle$ iff there exists an $\mathcal{E} \subseteq \mathcal{A}_\mathcal{P}$ called *explanation* such that $\mathcal{P} \cup \mathcal{E}$ is satisfiable, the least Ł-model of the weak completion of $\mathcal{P} \cup \mathcal{E}$ satisfies $\mathcal{IC}$, and $\mathcal{P} \cup \mathcal{E} \models_Ł^{lmwc} L$ for each $L \in \mathcal{O}$.

3 A Reduction System for Indicative Conditionals

When parsing conditionals we assume that information concerning the mood of the conditionals has been extracted. In this paper we restrict our attention to indicative mood. In the sequel let $\mathit{cond}(\mathcal{T}, \mathcal{A})$ be a conditional with condition $\mathcal{T}$

and consequence $\mathcal{A}$, both of which are assumed to be finite sets of literals not containing a complementary pair of literals, i.e., a pair B and $\neg B$.

Conditionals are evaluated wrt background information specified as a logic program and a set of integrity constraints. More specifically, as the weak completion of each logic program always admits a least Ł-model, the conditionals are evaluated under these least Ł-models. In the reminder of this section let $\mathcal{P}$ be a program, $\mathcal{IC}$ be a finite set of integrity constraints, and $\mathcal{M_P}$ be the least Ł-model of $wc\mathcal{P}$ such that $\mathcal{M_P}$ satisfies $\mathcal{IC}$. A *state* is either an expression of the form $ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A})$ or *true*, *false*, *unknown*, or *vacuous*.

3.1 A Revision Operator

Let $\mathcal{S}$ be a finite set of ground literals not containing a complementary pair of literals and let B be a ground atom in

$$rev(\mathcal{P}, \mathcal{S}) = (\mathcal{P} \setminus def(\mathcal{S}, \mathcal{P})) \cup \{B \leftarrow \top \mid B \in \mathcal{S}\} \cup \{B \leftarrow \bot \mid \neg B \in \mathcal{S}\}.$$

The revision operator ensures that all literals occurring in $\mathcal{S}$ are mapped to *true* under the least Ł-model of $wc\, rev(\mathcal{P}, \mathcal{S})$.

3.2 The Abstract Reduction System

Let $cond(\mathcal{T}, \mathcal{A})$ be an indicative conditional which is to be evaluated in the context of a logic program $\mathcal{P}$ and integrity constraints $\mathcal{IC}$ such that the least Ł-model $\mathcal{M_P}$ of $wc\mathcal{P}$ satisfies $\mathcal{IC}$. The initial state is $ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A})$.

If the condition of the conditional is *true*, then the conditional holds if its consequent is *true* as well; otherwise it is either *false* or *unknown*.

$$\begin{array}{lll}
ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{it} true & \text{iff} & \mathcal{M_P}(\mathcal{T}) = true \text{ and } \mathcal{M_P}(\mathcal{A}) = true \\
ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{if} false & \text{iff} & \mathcal{M_P}(\mathcal{T}) = true \text{ and } \mathcal{M_P}(\mathcal{A}) = false \\
ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{iu} unknown & \text{iff} & \mathcal{M_P}(\mathcal{T}) = true \text{ and } \mathcal{M_P}(\mathcal{A}) = unknown
\end{array}$$

If the condition of the conditional is *false*, then the conditional is *true* under Ł-semantics. However, we believe that humans might make a difference between a conditional whose condition and consequence is *true* and a conditional whose condition is *false*. Hence, for the time being we consider a conditional whose condition is *false* as *vacuous*.

$$ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{iv} vacuous \quad \text{iff} \quad \mathcal{M_P}(\mathcal{T}) = false$$

If the condition of the conditional is *unknown*, then we could assign a truth-value to the conditional in accordance with the Ł-semantics. However, we suggest that in this case abduction and revision shall be applied in order to satisfy the condition. We start with the *abduction rule*:

$$ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{ia} ic(\mathcal{P} \cup \mathcal{E}, \mathcal{IC}, \mathcal{T} \setminus \mathcal{O}, \mathcal{A})$$

iff $\mathcal{M}_\mathcal{P}(\mathcal{T}) = \mathit{unknown}$ and $\mathcal{E}$ explains $\mathcal{O} \subseteq \mathcal{T}$ in the abductive framework $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{L}^{lmwc} \rangle$ and $\mathcal{O} \neq \emptyset$. Please note that $\mathcal{T}$ may contain literals which are mapped to *true* by $\mathcal{M}_\mathcal{P}$. These literals can be removed from $\mathcal{T}$ by the rule $\longrightarrow_{ia}$ because the empty set explains them.

Now we turn to the *revision rule*:

$$ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A}) \longrightarrow_{ir} ic(rev(\mathcal{P}, \mathcal{S}), \mathcal{IC}, \mathcal{T} \setminus \mathcal{S}, \mathcal{A})$$

iff $\mathcal{M}_\mathcal{P}(\mathcal{T}) = \mathit{unknown}$, $\mathcal{S} \subseteq \mathcal{T}$, $\mathcal{S} \neq \emptyset$, for each $L \in \mathcal{S}$ we find $\mathcal{M}_\mathcal{P}(L) = \mathit{unknown}$, and the least Ł-model of $wc\, rev(\mathcal{P}, \mathcal{S})$ satisfies $\mathcal{IC}$.

Altogether we obtain the reduction system RIC operating on states and consisting of the rules $\{\longrightarrow_{it}, \longrightarrow_{if}, \longrightarrow_{iu}, \longrightarrow_{iv}, \longrightarrow_{ia}, \longrightarrow_{ir}\}$.

4 Examples

4.1 Al in the Jailhouse

Rainy Day Suppose we are told that Al is imprisoned in a jailhouse on a rainy day, i.e.., he is living in a cell inside the jailhouse and it is raining:

$$\mathcal{P}_1 = \{inside(X) \leftarrow imprisoned(X), imprisoned(al) \leftarrow \top, raining \leftarrow \top\}.$$

The least Ł-model of $wc\mathcal{P}_1$ is $\langle\{imprisoned(al), inside(al), raining\}, \emptyset\rangle$. In order to evaluate conditional (1) with respect to $\mathcal{P}_1$ we observe that this model maps *raining* and *inside* to *true*. Hence,

$$ic(\mathcal{P}_1, \emptyset, raining, inside) \longrightarrow_{it} true.$$

Sunny Day Let us assume that Al is still imprisoned but that it is not raining:

$$\mathcal{P}_2 = \{inside(X) \leftarrow imprisoned(X), imprisoned(al) \leftarrow \top, raining \leftarrow \bot\}.$$

The least Ł-model of $wc\mathcal{P}_2$ is $\langle\{imprisoned(al), inside(al)\}, \{raining\}\rangle$. In order to evaluate conditional (1) wrt $\mathcal{P}_2$ we observe that this model maps *raining* to *false*. Hence,

$$ic(\mathcal{P}_2, \emptyset, raining, inside) \longrightarrow_{iv} vacuous.$$

No Information about the Weather Suppose we are told that Al is imprisoned in a jailhouse but we know nothing about the weather:

$$\mathcal{P}_3 = \{inside(X) \leftarrow imprisoned(X), imprisoned(al) \leftarrow \top\}.$$

The least Ł-model of $wc\mathcal{P}_1$ is $\langle\{imprisoned(al), inside(al)\}, \emptyset\rangle$. In order to evaluate conditional (1) wrt $\mathcal{P}_3$ we observe that this model maps *raining* to *unknown*. Hence, we view *raining* as an observation which needs to be explained. The only possible explanation wrt $\langle \mathcal{P}_3, \{raining \leftarrow \top, raining \leftarrow \bot\}, \emptyset, \models_{L}^{lmwc} \rangle$ is $\{raining \leftarrow \top\}$. Altogether we obtain

$$ic(\mathcal{P}_3, \emptyset, raining, inside) \longrightarrow_{ia} ic(\mathcal{P}_1, \emptyset, \emptyset, inside) \longrightarrow_{it} true.$$

Please note that $\mathcal{P}_3 \cup \{raining \leftarrow \top\} = \mathcal{P}_1 = rev(\mathcal{P}_3, raining)$. Hence, we could replace $\longrightarrow_{ia}$ by $\longrightarrow_{ir}$ in the previous reduction sequence.

4.2 The Shooting of Kennedy

President Kennedy was killed. There was a lengthy investigation about who actually shot the president and in the end it was determined that Oswald did it:

$$\mathcal{P}_4 = \{\mathit{Kennedy\_dead} \leftarrow \mathit{os\_shot}, \mathit{Kennedy\_dead} \leftarrow \mathit{se\_shot}, \mathit{os\_shot} \leftarrow \top\}.$$

The least Ł-model of $wc\mathcal{P}_4$ is $\langle\{\mathit{os\_shot}, \mathit{Kennedy\_dead}\}, \emptyset\rangle$. Evaluating the indicative conditional (2) under this model we find that its condition $\mathcal{T} = \{\mathit{Kennedy\_dead}, \neg \mathit{os\_shot}\}$ is mapped to *false*. Hence,

$$ic(\mathcal{P}_4, \emptyset, \{\mathit{Kennedy\_dead}, \neg \mathit{os\_shot}\}, \mathit{se\_shot}) \longrightarrow_{iv} \mathit{vacuous}.$$

Now consider the case that we do not know that Oswald shot the president:

$$\mathcal{P}_5 = \{\mathit{Kennedy\_dead} \leftarrow \mathit{os\_shot}, \mathit{Kennedy\_dead} \leftarrow \mathit{se\_shot}\}.$$

As least Ł-model of $wc\mathcal{P}_5$ we obtain $\langle\emptyset, \emptyset\rangle$ and find that it maps $\mathcal{T}$ to *unknown*. We may try to consider $\mathcal{T}$ as an observation and explain it wrt the abductive framework $\langle\mathcal{P}_5, \mathcal{A}_{\mathcal{P}_5}, \emptyset, \models_{Ł}^{lmwc}\rangle$, where $\mathcal{A}_{\mathcal{P}_5}$ consists of the positive and negative facts for *os_shot* and *se_shot*. The only possible explanation is $\mathcal{E} = \{\mathit{os\_shot} \leftarrow \bot, \mathit{se\_shot} \leftarrow \top\}$. As least Ł-model of $wc(\mathcal{P}_5 \cup \mathcal{E})$ we obtain $\langle\{\mathit{Kennedy\_dead}, \mathit{se\_shot}\}, \{\mathit{os\_shot}\}\rangle$. As this model maps *se_shot* to *true* we find

$$\begin{aligned} & ic(\mathcal{P}_5, \emptyset, \{\mathit{Kennedy\_dead}, \neg \mathit{os\_shot}\}, \mathit{se\_shot}) \\ \longrightarrow_{ia}\ & ic(\mathcal{P}_5 \cup \mathcal{E}, \emptyset, \emptyset, \mathit{se\_shot}) \longrightarrow_{it} \mathit{true}. \end{aligned}$$

In this example we could also apply revision. Let

$$\mathcal{P}_6 = \mathit{rev}(\mathcal{P}_5, \mathcal{T}) = \{\mathit{Kennedy\_dead} \leftarrow \top, \mathit{os\_shot} \leftarrow \bot\}.$$

We obtain

$$\begin{aligned} & ic(\mathcal{P}_5, \emptyset, \{\mathit{Kennedy\_dead}, \neg \mathit{os\_shot}\}, \mathit{se\_shot}) \\ \longrightarrow_{ir}\ & ic(\mathcal{P}_6, \emptyset, \emptyset, \mathit{se\_shot}) \longrightarrow_{iu} \mathit{unknown} \end{aligned}$$

because the least Ł-model of $wc\mathcal{P}_6$ is $\langle\{\mathit{Kennedy\_dead}\}, \{\mathit{os\_shot}\}\rangle$ and maps *se_shot* to *unknown*. However, as conditional (2) can be evaluated by abduction and without revising the initial program, this derivation is not preferred.

4.3 The Firing Squad

This example is presented in [13]. If the court orders an execution, then the captain will give the signal upon which riflemen A and B will shoot the prisoner. Consequently, the prisoner will be dead. We assume that the court's decision is *unknown*, that both riflemen are accurate, alert and law-abiding, and that the prisoner is unlikely to die from any other causes. Let

$$\begin{aligned} \mathcal{P}_7 = \{\ & \mathit{sig} \leftarrow \mathit{execution}, \mathit{rmA} \leftarrow \mathit{sig}, \mathit{rmB} \leftarrow \mathit{sig}, \\ & \mathit{dead} \leftarrow \mathit{rmA}, \mathit{dead} \leftarrow \mathit{rmB}, \mathit{alive} \leftarrow \neg \mathit{dead}\ \}. \end{aligned}$$

The least Ł-model of $wc\mathcal{P}_7$ is

$$\langle\emptyset, \emptyset\rangle. \tag{7}$$

Rifleman A did not Shoot To evaluate conditional (3) wrt this model we first observe that the condition rmA is mapped to *unknown* by (7). Considering the abductive framework

$$\langle \mathcal{P}_7, \{execution \leftarrow \top, execution \leftarrow \bot\}, \emptyset, \models_{Ł}^{lmwc} \rangle, \tag{8}$$

$\neg rmA$ can be explained by

$$\{execution \leftarrow \bot\}. \tag{9}$$

Let $\mathcal{P}_8 = \mathcal{P}_7 \cup (9)$. The least Ł-model of $wc\mathcal{P}_8$ is

$$\langle \{alive\}, \{execution, sig, rmA, rmB, dead\} \rangle. \tag{10}$$

Because *alive* is mapped to *true* under this model, we obtain

$$ic(\mathcal{P}_7, \emptyset, \neg rmA, alive) \longrightarrow_{ia} ic(\mathcal{P}_8, \emptyset, \emptyset, alive) \longrightarrow_{it} true.$$

The Prisoner is Alive Now consider conditional (4). Because (7) maps *alive* to *unknown* we treat *alive* as an observation. Considering again the abductive framework (8) this observation can be explained by (9). Hence, we evaluate the consequence of (4) under (10) and find that the captain did not signal:

$$ic(\mathcal{P}_7, \emptyset, alive, \neg sig) \longrightarrow_{ia} ic(\mathcal{P}_8, \emptyset, \emptyset, \neg sig) \longrightarrow_{it} true.$$

Rifleman A Shot Let us turn the attention to conditional (5). Because (7) maps rmA to *unknown*, we treat rmA as an observation. Considering the abductive framework (8) this observation can be explained by

$$\{execution \leftarrow \top\}. \tag{11}$$

Let $\mathcal{P}_9 = \mathcal{P}_7 \cup (11)$. The least Ł-model of $wc\mathcal{P}_9$ is

$$\langle \{execution, sig, rmA, rmB, dead\}, \{alive\} \rangle. \tag{12}$$

Because rmB is mapped to *true* under this model, we obtain

$$ic(\mathcal{P}_7, \emptyset, rmA, rmB) \longrightarrow_{ia} ic(\mathcal{P}_9, \emptyset, \emptyset, rmB) \longrightarrow_{it} true.$$

The Captain Gave no Signal Let us now consider conditional (6). Its condition $\mathcal{T} = \{\neg sig, rmA\}$ is mapped to *unknown* by (7). We can only explain $\neg sig$ by (9) and rmA by (11), but we cannot explain $\mathcal{T}$ because

$$wc((9) \cup (11)) = \{execution \leftrightarrow \top \vee \bot\} \equiv \{execution \leftrightarrow \top\}.$$

In order to evaluate this conditional we have to consider revisions.

1. A brute force method is to revise the program wrt all conditions. Let

$$\begin{aligned}\mathcal{P}_{10} &= \mathit{rev}(\mathcal{P}_7, \{\neg \mathit{sig}, \mathit{rmA}\}) \\ &= (\mathcal{P}_7 \setminus \mathit{def}(\{\neg \mathit{sig}, \mathit{rmA}\}, \mathcal{P}_7)) \cup \{\mathit{sig} \leftarrow \bot, \mathit{rmA} \leftarrow \top\}.\end{aligned}$$

   The least Ł-model of $\mathit{wc}\mathcal{P}_{10}$ is

$$\langle \{\mathit{rmA}, \mathit{dead}\}, \{\mathit{sig}, \mathit{rmB}, \mathit{alive}\} \rangle. \tag{13}$$

   This model maps *dead* to *true* and *rmB* to *false* and we obtain

$$\begin{aligned}&\mathit{ic}(\mathcal{P}_7, \emptyset, \{\neg \mathit{sig}, \mathit{rmA}\}, \{\mathit{dead}, \neg \mathit{rmB}\}) \\ &\longrightarrow_{ir} \mathit{ic}(\mathcal{P}_{10}, \emptyset, \emptyset, \{\mathit{dead}, \neg \mathit{rmB}\}) \longrightarrow_{it} \mathit{true}.\end{aligned}$$

2. As we prefer minimal revisions let us consider

$$\mathcal{P}_{11} = \mathit{rev}(\mathcal{P}_7, \mathit{rmA}) = (\mathcal{P}_7 \setminus \mathit{def}(\mathit{rmA}, \mathcal{P}_7)) \cup \{\mathit{rmA} \leftarrow \top\}.$$

   The least Ł-model of $\mathit{wc}\mathcal{P}_{11}$ is $\langle \{\mathit{dead}, \mathit{rmA}\}, \{\mathit{alive}\} \rangle$. Unfortunately, $\neg \mathit{sig}$ is still mapped to *unknown* by this model, but it can be explained in the abductive framework $\langle \mathcal{P}_{11}, \{\mathit{execution} \leftarrow \top, \mathit{execution} \leftarrow \bot\}, \emptyset, \models_{Ł}^{lmwc} \rangle$ by (9). Let $\mathcal{P}_{12} = \mathcal{P}_{11} \cup (9)$. Because the least Ł-model of $\mathit{wc}\mathcal{P}_{12}$ is

$$\langle \{\mathit{dead}, \mathit{rmA}\}, \{\mathit{alive}, \mathit{execution}, \mathit{sig}, \mathit{rmB}\} \rangle \tag{14}$$

   we obtain

$$\begin{aligned}&\mathit{ic}(\mathcal{P}_7, \emptyset, \{\neg \mathit{sig}, \mathit{rmA}\}, \{\mathit{dead}, \neg \mathit{rmB}\}) \\ &\longrightarrow_{ir} \mathit{ic}(\mathcal{P}_{11}, \emptyset, \neg \mathit{sig}, \{\mathit{dead}, \neg \mathit{rmB}\}) \\ &\longrightarrow_{ia} \mathit{ic}(\mathcal{P}_{12}, \emptyset, \emptyset, \{\mathit{dead}, \neg \mathit{rmB}\} \qquad \longrightarrow_{it} \mathit{true}.\end{aligned}$$

   The revision leading to $\mathcal{P}_{11}$ is minimal in the sense that only the definition of *rmA* is revised and without this revision the condition of (6) cannot be explained. This is the only minimal revision as we will show in the sequel.
3. An alternative minimal revision could be the revision of $\mathcal{P}_7$ wrt to $\neg \mathit{sig}$:

$$\mathcal{P}_{13} = \mathit{rev}(\mathcal{P}_7, \neg \mathit{sig}) = (\mathcal{P}_7 \setminus \mathit{def}(\neg \mathit{sig}, \mathcal{P}_7)) \cup \{\mathit{sig} \leftarrow \bot\}.$$

   The least Ł-model of $\mathit{wc}\mathcal{P}_{13}$ is

$$\langle \{\mathit{alive}\}, \{\mathit{sig}, \mathit{dead}, \mathit{rmA}, \mathit{rmB}\} \rangle. \tag{15}$$

   Because this model maps *rmA* to *false* we obtain:

$$\begin{aligned}&\mathit{ic}(\mathcal{P}_7, \emptyset, \{\neg \mathit{sig}, \mathit{rmA}\}, \{\mathit{dead}, \neg \mathit{rmB}\}) \\ &\longrightarrow_{ir} \mathit{ic}(\mathcal{P}_{13}, \emptyset, \mathit{rmA}, \{\mathit{dead}, \neg \mathit{rmB}\}) \longrightarrow_{iv} \mathit{vacuous}.\end{aligned}$$

4. So far the first step in evaluating the conditional was a revision step. Alternatively, we could start with an abduction step. $\neg \mathit{sig}$ can be explained in the abductive framework (8) by (9) leading to the program $\mathcal{P}_8$ and the least Ł-model (10). Because this model maps *rmA* to *false* we obtain:

$$\begin{aligned}&\mathit{ic}(\mathcal{P}_7, \emptyset, \{\neg \mathit{sig}, \mathit{rmA}\}, \{\mathit{dead}, \neg \mathit{rmB}\}) \\ &\longrightarrow_{ia} \mathit{ic}(\mathcal{P}_8, \emptyset, \mathit{rmA}, \{\mathit{dead}, \neg \mathit{rmB}\}) \longrightarrow_{iv} \mathit{vacuous}.\end{aligned}$$

5. Let us now reverse the order in which the conditions are treated and start by explaining rmA. This has already been done before and we obtain $\mathcal{P}_9$ and the least Ł-model (12). Because this model maps $\neg sig$ to *false* we obtain:

$$\begin{array}{l} ic(\mathcal{P}_7, \emptyset, \{\neg sig, rmA\}, \{dead, \neg rmB\}) \\ \longrightarrow_{ia} ic(\mathcal{P}_9, \emptyset, \neg sig, \{dead, \neg rmB\}) \longrightarrow_{iv} vacuous. \end{array}$$

In the last example we have discussed five different approaches to handle the case that the truth value of the conditions of a conditional is *unknown* and cannot be explained: maximal (parallel) revision (MaxRev), partial (sequential) revision as well as partial (sequential) explanation, where in the sequential approaches the literals in the condition of the conditionals are treated in different orders: left-to-right and right-to-left, where we consider sets to be ordered (PRevLR,PRevRL,PExLR,PExRL). The results are summarized in Table 1, where the conditional as well as the literals are evaluated wrt the final least Ł-model computed in the different approaches.

Which approach shall be preferred? Because rifleman A causally depends on the captain's signal but not vice-versa, plus given that in this example clauses express causes, and effects come after causes, it would make sense to take the cause ordering as the preferred one for abducing the conditions. Hence, PExLR would be preferred. However, because rifleman A is an agent, the causes of his actions can be internal to him, his decisions. Hence, when autonomous agents are involved (or spontaneous phenomena like radioactivity), the ordering of abducing the conditions is independent of causal dependency.

5 Properties

In this section, let $\mathcal{P}$ be a program, $\langle I^\top, I^\bot \rangle$ the least Ł-model of $wc\mathcal{P}$, $\mathcal{IC}$ a set of integrity constraints, $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models^{lmwc}_{Ł} \rangle$ an abductive framework, and L a literal.

Proposition 1. *If $\mathcal{O}$ can be explained by $\mathcal{E} \subseteq \mathcal{A}_\mathcal{P}$ and $\langle J^\top, J^\bot \rangle$ is the least L-model of $wc(\mathcal{P} \cup \mathcal{E})$, then $\langle I^\top, I^\bot \rangle \subseteq \langle J^\top, J^\bot \rangle$.*

	MaxRev	PRevRL	PRevLR	PExLR	PExRL
final program	$\mathcal{P}_{10}$	$\mathcal{P}_{12}$	$\mathcal{P}_{13}$	$\mathcal{P}_8$	$\mathcal{P}_9$
final least Ł-model	(13)	(14)	(15)	(10)	(12)
sig	*false*	*false*	*false*	*false*	*true*
rmA	*true*	*true*	*false*	*false*	*true*
dead	*true*	*true*	*false*	*false*	*true*
rmB	*false*	*false*	*false*	*false*	*true*
alive	*false*	*false*	*true*	*true*	*false*
execution	*unknown*	*false*	*unknown*	*false*	*true*
conditional (6)	*true*	*true*	*vacuous*	*vacuous*	*vacuous*

Table 1. Different approaches to evaluate conditional (6).

Proof. The least Ł-models $\langle I^\top, I^\perp \rangle$ and $\langle J^\top, J^\perp \rangle$ are the least fixed points of the semantic operators $\Phi_\mathcal{P}$ and $\Phi_{\mathcal{P}\cup\mathcal{E}}$, respectively. Let $\langle I_n^\top, I_n^\perp \rangle$ and $\langle J_n^\top, J_n^\perp \rangle$ be the interpretations obtained after applying $\Phi_\mathcal{P}$ and $\Phi_{\mathcal{P}\cup\mathcal{E}}$ n-times to $\langle \emptyset, \emptyset \rangle$, respectively. We can show by induction on n that $\langle I_n^\top, I_n^\perp \rangle \subseteq \langle J_n^\top, J_n^\perp \rangle$. The proposition follows immediately. ■

Proposition 1 guarantees that whenever $\longrightarrow_{ia}$ is applied, previously checked conditions of a conditional need not to be re-checked. The following Proposition 3 gives the same guarantee whenever $\longrightarrow_{ir}$ is applied.

Proposition 2. *If the least Ł-model of wc$\mathcal{P}$ maps L to unknown and $\langle J^\top, J^\perp \rangle$ is the least Ł-model of wc $rev(\mathcal{P}, L)$, then $\langle I^\top, I^\perp \rangle \subset \langle J^\top, J^\perp \rangle$.*

Proof. By induction on the number of applications of $\Phi_\mathcal{P}$ and $\Phi_{rev(\mathcal{P},L)}$. ■

Proposition 3. RIC *is terminating.*

Proof. Each application of $\longrightarrow_{it}$, $\longrightarrow_{if}$, $\longrightarrow_{iu}$ or $\longrightarrow_{iv}$ leads to an irreducible expression. Let $cond(\mathcal{T}, \mathcal{A})$ be the conditional to which RIC is applied. Whenever $\longrightarrow_{ir}$ is applied then the definition of at least one literal L occurring in $\mathcal{T}$ is revised such that the least Ł-model of the weak completion of revised program maps L to *true*. Because $\mathcal{T}$ does not contain a complementary pair of literals this revised definition of L is never revised again. Hence, there cannot exist a rewriting sequence with infinitely many occurrences of $\longrightarrow_{ir}$. Likewise, there cannot exist a rewriting sequence with infinitely many occurrences of $\longrightarrow_{ia}$ because each application of $\longrightarrow_{ia}$ to a state $ic(\mathcal{P}, \mathcal{IC}, \mathcal{T}, \mathcal{A})$ reduces the number of literals occurring in the $\mathcal{T}$. ■

Proposition 4. RIC *is not confluent.*

Proof. This follows immediately from the examples presented in Section 4. ■

6 Open Questions and the Proposal of an Experiment

Open Questions The new approach gives rise to a number of questions. Which of the approaches is preferable? This may be a question of pragmatics imputable to the user. The default, because no pragmatic information has been added, is maximals revision for skepticism and minimal revisions for credulity. Do humans evaluate multiple conditions sequentially or in parallel? If multiple conditions are evaluated sequentially, are they evaluated by some preferred order? Shall explanations be computed skeptically or credulously? How can the approach be extended to handle subjunctive conditionals?

The Proposal of an Experiment Subjects are given the background information specified in the program $\mathcal{P}_9$. They are confronted with the conditionals like (6) as well as variants with different consequences (e.g., *execution* instead of $\{dead, \neg rmB\}$ or conditionals where the order of two conditions are reversed. We then ask the subjects to answer questions like: *Does the conditional hold?* or *Did the court order an execution?* Depending on the answers we may learn which approaches are preferred by humans.

Acknowledgements We thank Bob Kowalski for valuable comments on an earlier draft of the paper.

References

1. E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the abstract and the social case of the selection task. In *Proceedings Eleventh International Symposium on Logical Formalizations of Commonsense Reasoning*, 2013.
2. E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the suppression task. In N. Miyake, D. Peebles, and R. P. Cooper, editors, *Proceedings of the 34th Annual Conference of the Cognitive Science Society*, pages 1500–1505. Cognitive Science Society, 2012.
3. E.-A. Dietz, S. Hölldobler, and C. Wernhard. Modelling the suppression task under weak completion and well-founded semantics. *Journal of Applied Non-Classical Logics*, 24:61–85, 2014.
4. M. L. Ginsberg. Counterfactuals. *Artificial Intelligence*, 30(1):35–79, 1986.
5. S. Hölldobler and C. D. P. Kencana Ramli. Contraction properties of a semantic operator for human reasoning. In Lei Li and K. K. Yen, editors, *Proceedings of the Fifth International Conference on Information*, pages 228–231. International Information Institute, 2009.
6. S. Hölldobler and C. D. P. Kencana Ramli. Logic programs under three-valued Łukasiewicz's semantics. In P. M. Hill and D. S. Warren, editors, *Logic Programming*, volume 5649 of *Lecture Notes in Computer Science*, pages 464–478. Springer-Verlag Berlin Heidelberg, 2009.
7. S. Hölldobler and C. D. P. Kencana Ramli. Logics and networks for human reasoning. In C. Alippi, Marios M. Polycarpou, Christos G. Panayiotou, and Georgios Ellinasetal, editors, *Artificial Neural Networks – ICANN*, volume 5769 of *Lecture Notes in Computer Science*, pages 85–94. Springer-Verlag Berlin Heidelberg, 2009.
8. S. Hölldobler, T. Philipp, and C. Wernhard. An abductive model for human reasoning. In *Proceedings Tenth International Symposium on Logical Formalizations of Commonsense Reasoning*, 2011. commonsensereasoning.org/2011/proceedings.html.
9. A. C. Kakas, R. A. Kowalski, and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6):719–770, 1993.
10. D. Lewis. *Counterfactuals.* Blackwell Publishers, Oxford, 1973.
11. J. W. Lloyd. *Foundations of Logic Programming.* Springer, Berlin, Heidelberg, 1987.
12. J. Łukasiewicz. O logice trójwartościowej. *Ruch Filozoficzny*, 5:169–171, 1920. English translation: On Three-Valued Logic. In: *Jan Łukasiewicz Selected Works.* (L. Borkowski, ed.), North Holland, 87-88, 1990.
13. J. Pearl. *Causality: Models, Reasoning, and Inference.* Cambridge University Press, New York, USA, 2000.
14. L. M. Pereira and J. N. Aparício. Relevant counterfactuals. In *Proceedings 4th Portuguese Conference on Artificial Intelligence (EPIA)*, volume 390 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 1989.
15. L. M. Pereira, E.-A. Dietz, and S. Hölldobler. An abductive reasoning approach to the belief-bias effect. In C. Baral, G. De Giacomo, and T. Eiter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 14th International Conference*, pages 653–656, Cambridge, MA, 2014. AAAI Press.

16. L. M. Pereira, E.-A. Dietz, and S. Hölldobler. Contextual abductive reasoning with side-effects. In I. Niemelä, editor, *Theory and Practice of Logic Programming (TPLP)*, volume 14, pages 633–648, Cambridge, UK, 2014. Cambridge University Press.
17. F. Ramsey. *The Foundations of Mathematics and Other Logical Essays*. Harcourt, Brace and Company, 1931.
18. N. Rescher. *Conditionals*. MIT Press, Cambridge, MA, 2007.
19. K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science*. MIT Press, 2008.

The Logical Difference for EL: from Terminologies towards TBoxes*

Shasha Feng[1], Michel Ludwig[2], and Dirk Walther[2]

[1] Jilin University, China
fengss@jlu.edu.cn
[2] Theoretical Computer Science
TU Dresden, Germany
{michel,dirk}@tcs.inf.tu-dresden.de

Abstract. In this paper we are concerned with the logical difference problem between ontologies. The logical difference is the set of subsumption queries that follow from a first ontology but not from a second one. We revisit our solution to logical difference problem for $\mathcal{EL}$-terminologies based on finding simulations between hypergraph representations of the terminologies, and we investigate a possible extension of the method to general $\mathcal{EL}$-TBoxes.

1 Introduction

Ontologies are widely used to represent domain knowledge. They contain specifications of objects, concepts and relationships that are often formalised using a logic-based language over a vocabulary that is particular to an application domain. Ontology languages based on description logics [2] have been widely adopted, e.g., description logics are underlying the Web Ontology Language (OWL) and its profiles.[3] Numerous ontologies have already been developed, in particular, in knowledge intensive areas such as the biomedical domain, and they are made available in dedicated repositories such as the NCBO bioportal.[4]

Ontologies constantly evolve, they are regularly extended, corrected and refined. As the size of ontologies increases, their continued development and maintenance becomes more challenging as well. In particular, the need to have automated tool support for detecting and representing differences between versions of an ontology is growing in importance for ontology engineering.

The logical difference is taken to be the set of queries that produce different answers when evaluated over distinct versions of an ontology. The language and vocabulary of the queries can be adapted in such a way that exactly the differences of interest become visible, which can be independent of the syntactic representation of the ontologies. We consider ontologies formulated in the

* The second and third authors were supported by the German Research Foundation (DFG) within the Cluster of Excellence 'Center for Advancing Electronics Dresden'.

[3] http://www.w3.org/TR/owl2-overview/

[4] http://bioportal.bioontology.org

lightweight description logic $\mathcal{EL}$ [1, 3] and queries that are $\mathcal{EL}$-concept inclusions. The relevance of $\mathcal{EL}$ for ontologies is emphasised by the fact that many ontologies are largely formulated in $\mathcal{EL}$. For instance, the dataset of the ORE 2014 reasoner evaluation comprises 8 805 OWL-EL ontologies.[5]

The logical difference problem was introduced in [7] and investigated for $\mathcal{EL}$-terminologies [6]. A hypergraph-based approach for $\mathcal{EL}$-terminologies was presented in [4], which was subsequently extended to $\mathcal{EL}$-terminologies with additional role inclusions, domain and range restrictions of roles in [8]. In this paper we investigate a possible extension of the method to general $\mathcal{EL}$-TBoxes. Clearly, such an extension needs to account for the additional expressivity of general TBoxes w.r.t. terminologies. After normalisation, a terminology may contain at most one axiom of the form $\exists r.A \sqsubseteq X$ or $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq X$ for any concept name X, whereas a general TBox does not impose such a restriction.

We first show that for every concept inclusion $C \sqsubseteq D$ that follows from a TBox $\mathcal{T}$, there exists a concept name X in $\mathcal{T}$ that acts as an *interpolant* between the concepts C and D, i.e., we have that $\mathcal{T} \models C \sqsubseteq X$ and $\mathcal{T} \models X \sqsubseteq D$. Then we describe the set of all subsumees C of X in $\mathcal{T}$ using a concept of $\mathcal{EL}$ extended with disjunction and a least fixpoint operator, and the set of all subsumers D of X in $\mathcal{T}$ using a concept of $\mathcal{EL}$ extended with greatest fixpoint operators. Finally, we reduce the problem of deciding the logical difference between two $\mathcal{EL}$-TBoxes to fixpoint reasoning w.r.t. TBoxes in a hybrid μ-calculus [10].

The paper is organised as follows. We start by recalling some notions regarding the description logic $\mathcal{EL}$ and its extensions with disjunction and fixpoint operators. In Section 3, we discuss how the logical difference problem for $\mathcal{EL}$-terminologies could be extended to general $\mathcal{EL}$-TBoxes, and we establish a witness theorem for general $\mathcal{EL}$-TBoxes. In Section 4, we show how fixpoint reasoning can be used to decide whether two general $\mathcal{EL}$-TBoxes are logically different, and how witnesses to the logical difference can be computed. Finally we conclude the paper.

2 Preliminaries

We start by briefly reviewing the lightweight description logic $\mathcal{EL}$ and some notions related to the logical difference, together with some basic results.

Let $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_V}$ be mutually disjoint sets of concept names, role names, and variable names, respectively. We assume these sets to be countably infinite. We typically use A, B to denote concept names and r to denote role names.

The sets of $\mathcal{EL}$*-concepts* C, $\mathcal{ELU}_\mu$*-concepts* D, and $\mathcal{EL}_\nu$*-concepts* E are built according to the following grammar rules:

$$\begin{aligned}
C &::= \top \mid A \mid C \sqcap C \mid \exists r.C \\
D &::= \top \mid A \mid D \sqcap D \mid D \sqcup D \mid \exists r.D \mid x \mid \mu x.D \\
E &::= \top \mid A \mid E \sqcap E \mid \exists r.E \mid x \mid \nu x.E
\end{aligned}$$

[5] `http://dl.kr.org/ore2014/`

where $A \in \mathsf{N_C}$, $r, s \in \mathsf{N_R}$, and $x \in \mathsf{N_V}$. For an $\mathcal{ELU}_\mu$-concept C, the set of *free variables* in C, denoted by $\mathrm{FV}(C)$ is defined inductively as follows: $\mathrm{FV}(\top) = \emptyset$, $\mathrm{FV}(A) = \emptyset$, $\mathrm{FV}(D_1 \sqcap D_2) = \mathrm{FV}(D_1) \cup \mathrm{FV}(D_2)$, $\mathrm{FV}(D_1 \sqcup D_2) = \mathrm{FV}(D_1) \cup \mathrm{FV}(D_2)$, $\mathrm{FV}(\exists r.D') = \mathrm{FV}(D')$, $\mathrm{FV}(x) = \{x\}$, $\mathrm{FV}(\mu x.D') = \mathrm{FV}(D') \setminus \{x\}$. The set $\mathrm{FV}(E)$ of free variables occurring in an $\mathcal{EL}_\nu$-concept E can be defined analogously. An $\mathcal{ELU}_\mu$-concept C (an $\mathcal{EL}_\nu$-concept D) is *closed* if C (D) does not contain free occurrences of variables, i.e. $\mathrm{FV}(C) = \emptyset$ ($\mathrm{FV}(D) = \emptyset$). In the following we assume that every $\mathcal{ELU}_\mu$-concept C and every $\mathcal{EL}_\nu$-concept D is closed.

An *$\mathcal{EL}$-TBox $\mathcal{T}$* is a finite set of axioms, where an axiom can be a *concept inclusion* $C \sqsubseteq C'$, or a *concept equation* $C \equiv C'$, where C, C' range over $\mathcal{EL}$-concepts. An *$\mathcal{EL}$-terminology* $\mathcal{T}$ is an $\mathcal{EL}$-TBox consisting of axioms α of the form $A \sqsubseteq C$ and $A \equiv C$, where A is a concept name, C an $\mathcal{EL}$-concept and no concept name A occurs more than once on the left-hand side of an axiom.

The semantics of $\mathcal{EL}$, $\mathcal{ELU}_\mu$, and $\mathcal{EL}_\nu$ is defined using interpretations $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where the domain $\Delta^\mathcal{I}$ is a non-empty set, and $\cdot^\mathcal{I}$ is a function mapping each concept name A to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$ and every role name r to a binary relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. Interpretations are extended to concepts using a function $\cdot^{\mathcal{I},\xi}$ that is parameterised by an *assignment function* that maps variables $x \in \mathsf{N_V}$ to sets $\xi(x) \subseteq \Delta^\mathcal{I}$. Given an assignment ξ, the extension of an $\mathcal{EL}$, $\mathcal{ELU}_\mu$, or $\mathcal{EL}_\nu$-concept is defined inductively as follows: $\top^{\mathcal{I},\xi} := \Delta^\mathcal{I}$, $x^{\mathcal{I},\xi} := \xi(x)$ for $x \in \mathsf{N_V}$, $(C_1 \sqcap C_2)^{\mathcal{I},\xi} := C_1^\mathcal{I} \cap C_2^\mathcal{I}$, $(\exists r.C)^{\mathcal{I},\xi} := \{\, x \in \Delta^\mathcal{I} \mid \exists y \in C^{\mathcal{I},\xi} : (x,y) \in r^\mathcal{I} \,\}$, $(\mu x.C)^{\mathcal{I},\xi} = \bigcap\{\, W \subseteq \Delta^\mathcal{I} \mid C^{\mathcal{I},\xi[x \mapsto W]} \subseteq W \,\}$, and $(\nu x.C)^{\mathcal{I},\xi} = \bigcup\{\, W \subseteq \Delta^\mathcal{I} \mid W \subseteq C^{\mathcal{I},\xi[x \mapsto W]} \,\}$, where $\xi[x \mapsto W]$ denotes the assignment ξ modified by mapping x to W.

An interpretation $\mathcal{I}$ *satisfies* a concept C, an axiom $C \sqsubseteq D$ or $C \equiv D$ if, respectively, $C^{\mathcal{I},\emptyset} \neq \emptyset$, $C^{\mathcal{I},\emptyset} \subseteq D^\mathcal{I}$, or $C^{\mathcal{I},\emptyset} = D^{\mathcal{I},\emptyset}$. We write $\mathcal{I} \models \alpha$ iff $\mathcal{I}$ satisfies the axiom α. An interpretation $\mathcal{I}$ *satisfies* a TBox $\mathcal{T}$ iff $\mathcal{I}$ satisfies all axioms in $\mathcal{T}$; in this case, we say that $\mathcal{I}$ is a *model* of $\mathcal{T}$. An axiom α *follows* from a TBox $\mathcal{T}$, written $\mathcal{T} \models \alpha$, iff for all models $\mathcal{I}$ of $\mathcal{T}$, we have that $\mathcal{I} \models \alpha$. Deciding whether $\mathcal{T} \models C \sqsubseteq C'$, for two $\mathcal{EL}$-concepts C and C', can be done in polynomial time in the size of $\mathcal{T}$ and C, C' [1, 3]. For an $\mathcal{ELU}_\mu$-concept D and an $\mathcal{EL}_\nu$-concept E, it is known that $\mathcal{T} \models D \sqsubseteq E$ can be decided in exponential time in the size of $\mathcal{T}$, D and E [10].

A signature Σ is a finite set of symbols from $\mathsf{N_C}$ and $\mathsf{N_R}$. The signature $\mathsf{sig}(C)$, $\mathsf{sig}(\alpha)$ or $\mathsf{sig}(\mathcal{T})$ of the concept C, axiom α or TBox $\mathcal{T}$ is the set of concept and role names occurring in C, α or $\mathcal{T}$, respectively. An $\mathcal{EL}_\Sigma$-concept C is an $\mathcal{EL}$-concept such that $\mathsf{sig}(C) \subseteq \Sigma$.

An $\mathcal{EL}$-TBox $\mathcal{T}$ is *normalised* if it only contains $\mathcal{EL}$-concept inclusions of the forms $\top \sqsubseteq B$, $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B$, $A \sqsubseteq \exists r.B$, or $\exists r.A \sqsubseteq B$, where $A, A_i, B \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $n \geq 1$. Every $\mathcal{EL}$-TBox $\mathcal{T}$ can be normalised in polynomial time in the size of $\mathcal{T}$ with a linear increase in the size of the normalised TBox w.r.t. $\mathcal{T}$ such that the resulting TBox is a conservative extension of $\mathcal{T}$ [6]. Note that in a normalised terminology $\mathcal{T}$, we have that for every axiom of the form $\exists r.A \sqsubseteq B \in \mathcal{T}$, there exists an axiom of the form $B \sqsubseteq \exists r.A \in \mathcal{T}$; similarly for axioms of the form $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B$ with $n \geq 2$. When convenient, we will abbreviate two axioms $A \sqsubseteq \exists r.B$ and $\exists r.B \sqsubseteq A$ by the single axiom $A \equiv \exists r.B$; similarly for $A \equiv B_1 \sqcap \ldots \sqcap B_n$.

3 Towards Logical Difference between General $\mathcal{EL}$-TBoxes

The logical difference between two TBoxes witnessed by concept inclusions over a signature Σ is defined as follows.

Definition 1. *The* Σ-concept difference *between two $\mathcal{EL}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ for a signature Σ is the set* $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ *of all $\mathcal{EL}$-concept inclusions α such that* $\mathsf{sig}(\alpha) \subseteq \Sigma$, $\mathcal{T}_1 \models \alpha$, *and* $\mathcal{T}_2 \not\models \alpha$.

$\mathcal{EL}$-TBoxes can be translated into directed hypergraphs by taking the signature symbols as nodes and treating the axioms as hyperedges connecting the nodes. For normalised $\mathcal{EL}$-TBoxes, the axiom $\top \sqsubseteq B$ is translated into the hyperedge $(\{x_\top\}, \{x_B\})$, the axiom $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B$ into the hyperedge $(\{x_{B_1}, \ldots, x_{B_n}\}, \{x_A\})$, the axiom $A \sqsubseteq \exists r.B$ into the hyperedge $(\{x_A\}, \{x_r, x_B\})$, and the axiom $\exists r.A \sqsubseteq B$ into the hyperedge $(\{x_r, x_B\}, \{x_A\})$, where each node x_Y corresponds to the signature symbol Y, respectively. A feature of the translation of axioms into hyperedges is that all information about the axiom and the logical operators in it is preserved. In fact we can treat the ontology and its hypergraph representation interchangeably. The existence of certain simulations between hypergraphs for $\mathcal{EL}$-terminologies characterises the fact that the corresponding terminologies are logically equivalent and, thus, no logical difference exists [4,8].

As the set $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ is infinite in general, we make use of the following "primitive witnesses" theorem from [6] that states that we only have to consider two specific types of concept differences. If there is an inclusion $C \sqsubseteq D \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ for two terminologies $\mathcal{T}_1$ and $\mathcal{T}_2$, then we know that there is a concept name $A \in \Sigma$ such that A occurs either on the left-hand or the right-hand side of an inclusion in the set $C \sqsubseteq D \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$. For checking whether $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$, we only have to consider such simple inclusions. However, if $\mathcal{T}_1$ and $\mathcal{T}_2$ are general $\mathcal{EL}$-TBoxes, the situation is different.

Example 1. Let $\mathcal{T}_1 = \{X \equiv A_1 \sqcap A_2, X \sqsubseteq \exists r.\top\}$, $\mathcal{T}_2 = \emptyset$, and let $\Sigma = \{A_1, A_2, r\}$. Note that $\mathcal{T}_1$ is not a terminology as the concept name X occurs twice on the left-hand side of an axiom. Then every difference $\alpha \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ is equivalent to the inclusion $A_1 \sqcap A_2 \sqsubseteq \exists r.\top$. In particular, there does not exist a difference of the form $\psi \sqsubseteq \theta$, where ψ or θ is a concept name from Σ.

As illustrated by the example, we need to account for a new kind of differences $C \sqsubseteq C' \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ which are induced by a concept name $X \in \mathsf{sig}(\mathcal{T}_1)$ such that $X \notin \Sigma$, $\mathcal{T}_1 \models C \sqsubseteq X$, and $\mathcal{T}_1 \models X \sqsubseteq C'$. We obtain the following witness theorem for $\mathcal{EL}$-TBoxes as an extension of the witness theorem for $\mathcal{EL}$-terminologies.

Theorem 1 (Witness Theorem). *Let $\mathcal{T}_1$, $\mathcal{T}_2$ be two normalised $\mathcal{EL}$-TBoxes and let Σ be a signature. Then, $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) \neq \emptyset$ iff there exists an $\mathcal{EL}_\Sigma$-inclusion $\alpha = \varphi \sqsubseteq \psi$ such that $\mathcal{T}_1 \models \alpha$ and $\mathcal{T}_2 \not\models \alpha$, where*

(i) φ is an $\mathcal{EL}_\Sigma$-concept and $\psi = A \in \Sigma$,
(ii) $\varphi = A \in \Sigma$ and ψ is an $\mathcal{EL}_\Sigma$-concept, or
(iii) there exists $X \in \mathsf{sig}(\mathcal{T}_1) \setminus \Sigma$ such that $\mathcal{T}_1 \models \varphi \sqsubseteq X$ and $\mathcal{T}_1 \models X \sqsubseteq \psi$.

The proof of the witness theorem for terminologies [6] is based on analysing the subsumption $\mathcal{T}_1 \models \varphi \sqsubseteq \psi$ syntactically, using a sequent calculus [5]. A similar technique can be used for the proof of Theorem 2.

For deciding whether $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ in the case of general TBoxes, we now have to additionally consider differences of Type (*iii*). Differences of types (*i*) or (*ii*) can be checked by using forward or backward simulations adapted to normalised $\mathcal{EL}$-TBoxes, respectively, whereas Type (*iii*) differences require a combination of both techniques.

Before we illustrate how Type (*iii*) differences can be dealt with, we first introduce some auxiliary notions. We define $\mathsf{cWtn}^{\mathsf{lhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ as the set of all concept names A from Σ such that there exists an $\mathcal{EL}_\Sigma$-concept C with $A \sqsubseteq C \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$. Similarly, $\mathsf{cWtn}^{\mathsf{rhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ is the set of all concept names $A \in \Sigma$ such that there exists an $\mathcal{EL}_\Sigma$-concept C with $C \sqsubseteq A \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$. The concept names in $\mathsf{cWtn}^{\mathsf{lhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ are called *left-hand side witnesses* and the concept names in $\mathsf{cWtn}^{\mathsf{rhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ *right-hand side witnesses*. Additionally, we define $\mathsf{cWtn}^{\mathsf{mid}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ as the set of all concept names X from $\mathsf{sig}(\mathcal{T}_1)$ but not from Σ such that there exists $C \sqsubseteq C' \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ and $\mathcal{T}_1 \models C \sqsubseteq X$ and $\mathcal{T}_1 \models X \sqsubseteq C'$. The concept names in $\mathsf{cWtn}^{\mathsf{mid}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ are called *interpolating witnesses*. To summarise, we have the following sets:

$$
\begin{aligned}
\mathsf{cWtn}^{\mathsf{lhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) &= \{\, A \in \Sigma \mid \exists C \in \mathcal{EL}_\Sigma \colon A \sqsubseteq C \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) \,\} \\
\mathsf{cWtn}^{\mathsf{rhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) &= \{\, A \in \Sigma \mid \exists C \in \mathcal{EL}_\Sigma \colon C \sqsubseteq A \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) \,\} \\
\mathsf{cWtn}^{\mathsf{mid}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) &= \{\, X \in \mathsf{sig}(\mathcal{T}_1) \setminus \Sigma \mid \exists C, C' \in \mathcal{EL}_\Sigma \colon \mathcal{T}_1 \models C \sqsubseteq X, \\
&\qquad\qquad\qquad\qquad\qquad\qquad \mathcal{T}_1 \models X \sqsubseteq C', \mathcal{T}_2 \not\models C \sqsubseteq C' \,\}
\end{aligned}
$$

We illustrate the witness sets with the following example.

Example 2. Let $\mathcal{T}_1 = \{X \equiv A_1 \sqcap A_2, X \sqsubseteq \exists r.\top, A_3 \sqsubseteq A_2, A_3 \sqsubseteq \exists r.A_2\}$, $\mathcal{T}_2 = \{A_2 \sqsubseteq \exists r.\top\}$, and let $\Sigma = \{A_1, A_2, r\}$. Then it holds that $\mathsf{cWtn}^{\mathsf{lhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \{A_3\}$ (e.g. $\{A_3 \sqsubseteq A_2, A_3 \sqsubseteq \exists r.A_2\} \subseteq \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$), $\mathsf{cWtn}^{\mathsf{rhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \{A_2\}$ (e.g. $A_3 \sqsubseteq A_2 \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$), and $\mathsf{cWtn}^{\mathsf{mid}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \{X\}$ (e.g. $\mathcal{T}_1 \models A_1 \sqcap A_3 \sqsubseteq X$, $\mathcal{T}_1 \models X \sqsubseteq \exists r.\top$ and $\mathcal{T}_2 \not\models A_1 \sqcap A_3 \sqsubseteq \exists r.\top$).

We obtain as a corollary of Theorem 2 that, to decide the logical difference between two $\mathcal{EL}$-TBoxes, it is sufficient to check the emptiness of the witness sets.

Corollary 1. *Let $\mathcal{T}_1, \mathcal{T}_2$ be two normalised $\mathcal{EL}$-TBoxes and let Σ be a signature. Then it holds that $\mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ iff $\mathsf{cWtn}^{\mathsf{lhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \mathsf{cWtn}^{\mathsf{rhs}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \mathsf{cWtn}^{\mathsf{mid}}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.*

To characterise an interpolating witness of a Type-(*iii*) difference, we use the sets of its of subsumees and subsumers formulated using certain signature symbols only. A similar approach was used for the construction of uniform interpolants of $\mathcal{EL}$-TBoxes in [9].

Definition 2. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, let Σ be a signature and let C be an $\mathcal{EL}$-concept. We define* $\mathrm{Premises}_{\mathcal{T}}^{\Sigma}(C) := \{\, E \in \mathcal{EL}_{\Sigma} \mid \mathcal{T} \models E \sqsubseteq C \,\}$ *and* $\mathrm{Conclusions}_{\mathcal{T}}^{\Sigma}(C) := \{\, E \in \mathcal{EL}_{\Sigma} \mid \mathcal{T} \models C \sqsubseteq E \,\}$.

The set $\mathrm{Premises}_{\mathcal{T}}^{\Sigma}(C)$ contains all $\mathcal{EL}$-concepts formulated using Σ-symbols only that entail C w.r.t. $\mathcal{T}$; or are entailed by C in the case of $\mathrm{Conclusions}_{\mathcal{T}}^{\Sigma}(C)$. The elements of $\mathrm{Premises}_{\mathcal{T}}^{\Sigma}(C)$ are also called Σ-implicants or Σ-subsumees of C w.r.t. $\mathcal{T}$, and the elements of $\mathrm{Conclusions}_{\mathcal{T}}^{\Sigma}(C)$ are also named Σ-implicates or Σ-subsumers of C w.r.t. $\mathcal{T}$.

In [4], it was established that a concept name X is forward simulated by a concept name Y in an $\mathcal{EL}$-terminology $\mathcal{T}$ iff it holds that $\mathrm{Conclusions}_{\mathcal{T}}^{\Sigma}(X) \subseteq \mathrm{Conclusions}_{\mathcal{T}}^{\Sigma}(Y)$; and similary, X is backward simulated by Y iff $\mathrm{Premises}_{\mathcal{T}}^{\Sigma}(X) \subseteq \mathrm{Premises}_{\mathcal{T}}^{\Sigma}(Y)$. We aim now at lifting this result to general $\mathcal{EL}$-TBoxes.

Example 3. Let $\mathcal{T}_1 = \{A \sqsubseteq X, \exists r.X \sqsubseteq X, X \sqsubseteq B_1, X \sqsubseteq B_2\}$ and $\mathcal{T}_2 = \{A \sqsubseteq Y, \exists r.Y \sqsubseteq Y', \exists r.Y' \sqsubseteq Y, \exists r.Y \sqsubseteq Z_1, \exists r.Y \sqsubseteq Z_2, Y \sqsubseteq B_1, Y \sqsubseteq B_2, Z_1 \sqsubseteq B_1, Z_2 \sqsubseteq B_2\}$ be two $\mathcal{EL}$-TBoxes. Let $\Sigma = \{A, B_1, B_2, r\}$ be a signature. Note that X in $\mathcal{T}_1$ is cyclic and intuitively, the interpretation of X in a model $\mathcal{I}$ of $\mathcal{T}_1$ contains all finite r-chains "ending in A". In $\mathcal{T}_2$ the concept name Y is cyclic and its interpretation contains all r-chains ending in A that are of *even* length, whereas the interpretations of the concept names Z_1 and Z_2 contain all r-chains ending in A that are of *odd* length. Formally, we have that:

$$\begin{aligned} \{A, \exists r.A, \exists r.\exists r.A, \dots\} &\subseteq \mathrm{Premises}_{\mathcal{T}_1}^{\Sigma}(X) \\ \{A,\ \exists r.\exists r.A,\ \exists r.\exists r.\exists r.\exists r.A, \dots\} &\subseteq \mathrm{Premises}_{\mathcal{T}_2}^{\Sigma}(Y) \\ \{\exists r.A,\ \exists r.\exists r.\exists r.A, \dots\} &\subseteq \mathrm{Premises}_{\mathcal{T}_2}^{\Sigma}(Z_i) \text{ for } i \in \{1,2\} \end{aligned}$$

In particular, for $i \in \{1,2\}$, we have

$$\mathrm{Premises}_{\mathcal{T}_1}^{\Sigma}(X) = \mathrm{Premises}_{\mathcal{T}_2}^{\Sigma}(Y) \cup \mathrm{Premises}_{\mathcal{T}_2}^{\Sigma}(Z_i).$$

Intuitively, the set of Σ-implicants of X in $\mathcal{T}_1$ are distributed over the concept names Y and Z_i in $\mathcal{T}_2$. Moreover it holds that

$$\mathrm{Conclusions}_{\mathcal{T}_1}^{\Sigma}(X) = \mathrm{Conclusions}_{\mathcal{T}_2}^{\Sigma}(Y) = \mathrm{Conclusions}_{\mathcal{T}_2}^{\Sigma}(Z_1 \sqcap Z_2).$$

The concept name X in $\mathcal{T}_1$ could be forward simulated either by Y or $Z_1 \sqcap Z_2$ in $\mathcal{T}_2$. Note that Z_1 or Z_2 individually are not sufficient. Analogously, X could be backward simulated by $Y \sqcup Z_1$ or $Y \sqcup Z_2$. None of the concept names X, Z_1, or Z_2 are sufficient individually for the backward simulation. Combining backward and forward simulation, X could be simulated by $Y \sqcup (Z_1 \sqcap Z_2)$.

In general, we hypothesise that non-Σ-concept names X in $\mathcal{T}_1$ need to be "simulated" by concepts of the form $\bigsqcup_{i=1}^{n} C_i$, where C_i are $\mathcal{EL}$-concepts.

4 Finding Logical Differences via Fixpoint Reasoning

We now show how fixpoint reasoning can be used to find difference witnesses between general $\mathcal{EL}$-TBoxes.

Given Theorem 2, we know that any difference $C \sqsubseteq C' \in \mathsf{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$, for two $\mathcal{EL}_\Sigma$-concepts C and C', is connected to some concept name X occurring in $\mathcal{T}_1$ for which either $\mathcal{T}_1 \models C \sqsubseteq X$, or $\mathcal{T}_1 \models X \sqsubseteq C'$ (or both) holds. To check whether X is indeed a difference witness, we construct concepts $\mathrm{B}^\Sigma_{\mathcal{T}_1}(X)$ and $\mathrm{F}^\Sigma_{\mathcal{T}_1}(X)$ formulated in $\mathcal{ELU}^\Sigma_\mu$ and in $\mathcal{EL}^\Sigma_\nu$, respectively, to describe the (potentially infinite) disjunction of Σ-concepts that are subsumed by X w.r.t. $\mathcal{T}_1$, and the conjunction of all the Σ-concepts that subsume X w.r.t. $\mathcal{T}_1$, respectively. Note that the use of fixpoint allows for a finite description of infinite disjunctions or conjunctions. The $\mathcal{ELU}^\Sigma_\mu$-concept $\mathrm{B}^\Sigma_{\mathcal{T}_1}(X)$ hence is a finite representation of the set $\mathrm{Premises}^\Sigma_{\mathcal{T}_1}(X)$, whereas the $\mathcal{EL}^\Sigma_\nu$-concept $\mathrm{F}^\Sigma_{\mathcal{T}_1}(X)$ represents the set $\mathrm{Premises}^\Sigma_{\mathcal{T}_1}(X)$ in a finite way. Using the fixpoint descriptions of the premises and conclusions of X w.r.t. $\mathcal{T}_1$, we can verify whether X is a difference witness by checking $\mathcal{T}_2 \models \mathrm{B}^\Sigma_{\mathcal{T}_1}(X) \sqsubseteq \mathrm{F}^\Sigma_{\mathcal{T}_1}(X)$.

We first turn our attention to the set $\mathrm{Premises}^\Sigma_{\mathcal{T}_1}(X)$. Before we can give a formal definition for the concept $\mathrm{B}^\Sigma_{\mathcal{T}_1}(X)$, we have to introduce the following auxiliary notion to handle concept names X in the definition of $\mathrm{B}^\Sigma_{\mathcal{T}_1}(X)$ for which there exist axioms of the form $Z_1 \sqcap \ldots \sqcap Z_n \sqsubseteq Z$ in a normalised TBox $\mathcal{T}$ such that $\mathcal{T} \models Z \sqsubseteq X$. Intuitively, given a concept name X, we construct a set $\mathrm{Conj}_\mathcal{T}(X)$ containing sets of concept names which has the property that for every $\mathcal{EL}$-concept D with $\mathcal{T} \models D \sqsubseteq X$, there exists a set $S = \{Y_1, \ldots, Y_m\} \in \mathrm{Conj}_\mathcal{T}(X)$ such that $\mathcal{T} \models D \sqsubseteq Y_i$ follows without involving any axioms of the form $Z_1 \sqcap \ldots \sqcap Z_n \sqsubseteq Z$. Nested implications between such axioms also have to be taken into account.

Definition 3. *Let $\mathcal{T}$ be a normalised $\mathcal{EL}$-TBox and let $X \in \mathsf{N_C}$. We define the set $Conj_\mathcal{T}(X) \subseteq 2^{\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C}}$ to be smallest set inductively defined as follows:*

- *$\{X\} \in Conj_\mathcal{T}(X)$;*
- *if $S \in Conj_\mathcal{T}(X)$, $Y \in S$, and $Z_1 \sqcap \ldots \sqcap Z_n \sqsubseteq Z \in \mathcal{T}$ such that $n \geq 2$ and $\mathcal{T} \models Z \sqsubseteq Y$, then $S \setminus \{Y\} \cup \{Z_1, \ldots, Z_n\} \in Conj_\mathcal{T}(X)$.*

Note that for every concept name X the set $\mathrm{Conj}_\mathcal{T}(X)$ is finite as $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C}$ is finite.

Example 4. Let $\mathcal{T}_1 = \{A \sqsubseteq X, \exists r.X \sqsubseteq X\}$. Then $\mathrm{Conj}_{\mathcal{T}_1}(X) = \{\{X\}\}$. For $\mathcal{T}_2 = \{X_1 \sqcap X_2 \sqsubseteq X,\ X_3 \sqcap X_4 \sqsubseteq X_1,\ Y_1 \sqcap Y_2 \sqsubseteq X\}$, we have that

$$\mathrm{Conj}_{\mathcal{T}_2}(X) = \{\{X\}, \{X_1, X_2\}, \{X_3, X_4, X_2\}, \{Y_1, Y_2\}\}.$$

We can now give a formal definition of the concept $\mathrm{B}^\Sigma_\mathcal{T}(X)$.

Definition 4. *Let $\mathcal{T}$ be a normalised $\mathcal{EL}$-TBox, let Σ be a signature, and let $X \in \mathsf{sig}(\mathcal{T})$. For a mapping $\eta\colon \mathsf{N_C} \to \mathsf{N_V}$, we define a closed $\mathcal{ELU}^\mu_\Sigma$-concept $B^\Sigma_\mathcal{T}(X, \eta)$ as follows. We set $B^\Sigma_\mathcal{T}(X, \eta) = \top$ if $\mathcal{T} \models \top \sqsubseteq X$; otherwise $B^\Sigma_\mathcal{T}(X, \eta)$ is defined recursively in the following way:*

– *If* $X \in \mathsf{dom}(\eta)$*, then*

$$B_{\mathcal{T}}^{\Sigma}(X, \eta) = \eta(X)$$

– *If* $X \notin \mathsf{dom}(\eta)$*, we set*

$$B_{\mathcal{T}}^{\Sigma}(X, \eta) = \mu x. \bigsqcup_{\substack{S \in Conj_{\mathcal{T}}(X) \\ S = \{Y_1, \ldots, Y_m\}}} (Y_1' \sqcap \ldots \sqcap Y_m')$$

where x *is a fresh variable, and* Y_i' $(1 \leq i \leq n)$ *is defined as follows for* $\eta' := \eta \cup \{X \mapsto x\}$*:*

$$Y_i' = \bigsqcup_{\substack{\mathcal{T} \models B \sqsubseteq Y_i \\ B \in \Sigma}} B \sqcup \bigsqcup_{\substack{\exists r.Z \sqsubseteq Y \in \mathcal{T} \\ r \in \Sigma \\ \mathcal{T} \models Y \sqsubseteq Y_i}} \exists r. B_{\mathcal{T}}^{\Sigma}(Z, \eta')$$

Finally, we set $B_{\mathcal{T}}^{\Sigma}(X) = B_{\mathcal{T}}^{\Sigma}(X, \emptyset)$*.*

Intuitively, the construction of $\mathrm{B}_{\mathcal{T}}^{\Sigma}(X)$ starts from X and recursively collects all the concept names contained in Σ and all the left-hand sides of axioms in $\mathcal{T}$ that could be relevant for X to be entailed by a concept w.r.t. $\mathcal{T}$. By taking into account all possible axioms that could lead to a logical entailment, it is guaranteed that we capture every Σ-concept from which X follows w.r.t. $\mathcal{T}$. Reasoning involving axioms of the form $Z_1 \sqcap \ldots \sqcap Z_n \sqsubseteq Z$ is handled by the set $\mathrm{Conj}_{\mathcal{T}}(X)$. Infinite recursion over concepts of the form $\exists r.C$ is avoided by keeping track of the concept names that been visited already using the mapping η.

We note that for a normalised $\mathcal{EL}$-terminology $\mathcal{T}$, the concept $\mathrm{B}_{\mathcal{T}}^{\Sigma}(X)$ is of a simpler form than for normalised $\mathcal{EL}$-TBoxes. This is because the concept name X can occur on the right-hand side of at most one axiom of the form $\exists r.A \sqsubseteq X$ or $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq X$ with $n \geq 2$ in $\mathcal{T}$, whereas in a TBox several such axioms may occur.

We illustrate the concept $\mathrm{B}_{\mathcal{T}}^{\Sigma}(X)$ with the following examples.

Example 5. Let $\mathcal{T}_1 = \{A_1 \sqcap A_2 \sqsubseteq X,\ A_3 \sqsubseteq A_2, \exists r.A_2 \sqsubseteq A_1, \exists r.A_2 \sqsubseteq X\}$, $\mathcal{T}_2 = \mathcal{T}_1 \cup \{\exists r.X \sqsubseteq A_2\}$, and let $\Sigma = \{A_1, A_2, A_3, r\}$. We obtain the following $\mathcal{ELU}_{\Sigma}^{\mu}$-concepts. We write φ instead of $\mu x.\varphi$ if x does not occur freely in φ.

$$\begin{aligned}
\mathrm{B}_{\mathcal{T}_1}^{\Sigma}(A_1) &= A_1 \sqcup \exists r.(A_2 \sqcup A_3) \qquad \mathrm{B}_{\mathcal{T}_1}^{\Sigma}(A_2) = A_2 \sqcup A_3 \\
\mathrm{B}_{\mathcal{T}_1}^{\Sigma}(X) &= ((A_1 \sqcup \exists r.(A_2 \sqcup A_3)) \sqcap (A_2 \sqcup A_3)) \sqcup \exists r.(A_2 \sqcup A_3) \\
\mathrm{B}_{\mathcal{T}_2}^{\Sigma}(X) &= \mu x.(((A_1 \sqcup \exists r.(A_2 \sqcup A_3 \sqcup \exists r.x)) \sqcap (A_2 \sqcup A_3 \sqcup \exists r.x)) \\
&\qquad \sqcup \exists r.(A_2 \sqcup A_3 \sqcup \exists r.x))
\end{aligned}$$

Example 6. Let $\mathcal{T}_1$, $\mathcal{T}_2$ be defined as in Example 3, and let $\Sigma = \{A, B_1, B_2, r\}$. We have that for $i \in \{1, 2\}$:

$$\begin{aligned}
\mathrm{B}_{\mathcal{T}_1}^{\Sigma}(X) &= \mu x.(A \sqcup \exists r.x) \qquad \mathrm{B}_{\mathcal{T}_1}^{\Sigma}(B_i) = B_i \sqcup A \sqcup \exists r.\mu x.(A \sqcup \exists r.x) \\
\mathrm{B}_{\mathcal{T}_2}^{\Sigma}(Y) &= \mu y.(A \sqcup \exists r.\exists r.y) \qquad \mathrm{B}_{\mathcal{T}_2}^{\Sigma}(Z_i) = \exists r.\mu y.(A \sqcup \exists r.\exists r.y) \\
\mathrm{B}_{\mathcal{T}_2}^{\Sigma}(B_i) &= B_i \sqcup A \sqcup \exists r.\mu y_1.(\exists r.(A \sqcup \exists r.y_1)) \sqcup \exists r.\mu y_2.(A \sqcup \exists r.\exists r.y_2)
\end{aligned}$$

By inspecting Definition 4 it is easy to see that $\models \mathrm{B}^{\Sigma}_{\mathcal{T}}(X) \equiv \bot$ if there does not an $\mathcal{EL}_{\Sigma}$-concept C with $\mathcal{T} \models C \sqsubseteq X$. Overall, one can establish the following correctness and completeness properties.

Lemma 1. *Let $\mathcal{T}$ be a normalised $\mathcal{EL}$-TBox, let Σ be a signature, and let $X \in \mathsf{sig}(\mathcal{T})$. Then the $\mathcal{ELU}^{\Sigma}_{\mu}$-concept $B^{\Sigma}_{\mathcal{T}}(X)$ satisfies the following properties:*

(i) $\mathcal{T} \models B^{\Sigma}_{\mathcal{T}}(X) \sqsubseteq X$, *and*
(ii) *for every* $D \in \mathrm{Premises}^{\Sigma}_{\mathcal{T}}(X)$,

$$\mathcal{T} \models D \sqsubseteq X \text{ iff } \models D \sqsubseteq B^{\Sigma}_{\mathcal{T}}(X).$$

The following lemma states that the $\mathcal{ELU}_{\mu}$-concept $\mathrm{B}^{\Sigma}_{\mathcal{T}}(X)$ exactly captures the infinite set $\mathrm{Premises}^{\Sigma}_{\mathcal{T}}(X)$. More formally, the concept $\mathrm{B}^{\Sigma}_{\mathcal{T}}(X)$ is equivalent to the infinite disjunction over all the concepts contained in the set $\mathrm{Premises}^{\Sigma}_{\mathcal{T}}(X)$.

Lemma 2. *Let $\mathcal{T}$ be a normalised $\mathcal{EL}$-TBox, let Σ be a signature, and let $X \in \mathsf{sig}(\mathcal{T})$. Then for every interpretation $\mathcal{I}$ it holds that*

$$(B^{\Sigma}_{\mathcal{T}}(X))^{\mathcal{I},\emptyset} = \bigcup \{\, C^{\mathcal{I},\emptyset} \mid C \in \mathrm{Premises}^{\Sigma}_{\mathcal{T}}(X)\,\}.$$

Analogously to the concept $\mathrm{B}^{\Sigma}_{\mathcal{T}}(X)$, it is possible to construct an $\mathcal{EL}^{\Sigma}_{\nu}$-concept $\mathrm{F}^{\Sigma}_{\mathcal{T}}(X)$ which exactly captures the set $\mathrm{Conclusions}^{\Sigma}_{\mathcal{T}}(X)$ for a concept name X and an $\mathcal{EL}$-TBox $\mathcal{T}$. Due to lack of space, we cannot give a full definition of the concept $\mathrm{F}^{\Sigma}_{\mathcal{T}}(X)$. Instead, we state its existence and its essential property in the following lemma.

Lemma 3. *Let $\mathcal{T}$ be a normalised $\mathcal{EL}$-TBox, let Σ be a signature, and let $X \in \mathsf{sig}(\mathcal{T})$. Then there exists an $\mathcal{EL}_{\nu}$-concept $F^{\Sigma}_{\mathcal{T}}(X)$ such that for every interpretation $\mathcal{I}$ it holds that*

$$(F^{\Sigma}_{\mathcal{T}}(X))^{\mathcal{I},\emptyset} = \bigcup \{\, C^{\mathcal{I},\emptyset} \mid C \in \mathrm{Conclusions}^{\Sigma}_{\mathcal{T}}(X)\,\}.$$

We can now state the following theorem, which establishes how the concepts $\mathrm{B}^{\Sigma}_{\mathcal{T}}(X)$ and $\mathrm{F}^{\Sigma}_{\mathcal{T}}(X)$ can be used to search for difference witnesses.

Theorem 2. *Let $\mathcal{T}_1$, $\mathcal{T}_2$ be two normalised $\mathcal{EL}$-TBoxes. Then it holds that:*

(i) $A \notin \mathsf{cWtn}^{\mathsf{lhs}}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ *iff* $\mathcal{T}_2 \models A \sqsubseteq F^{\Sigma}_{\mathcal{T}_1}(A)$, *for every* $A \in \Sigma$;
(ii) $A \notin \mathsf{cWtn}^{\mathsf{rhs}}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ *iff* $\mathcal{T}_2 \models B^{\Sigma}_{\mathcal{T}_1}(A) \sqsubseteq A$, *for every* $A \in \Sigma$; *and*
(iii) $X \notin \mathsf{cWtn}^{\mathsf{mid}}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ *iff* $\mathcal{T}_2 \models B^{\Sigma}_{\mathcal{T}_1}(X) \sqsubseteq F^{\Sigma}_{\mathcal{T}_1}(X)$, *for every* $X \in \mathsf{sig}(\mathcal{T}_1) \setminus \Sigma$.

Theorem 2 together with Corollary 1 give rise to an algorithm for deciding the logical difference between $\mathcal{EL}$-TBoxes. Procedure 1 is such an algorithm based on reasoning in the hybrid μ-calculus, which allows for fixpoint reasoning w.r.t. TBoxes [10].

Theorem 3. *Procedure 1 runs in ExpTime.*

Procedure 1 Deciding existence of logical difference

Input: Normalised $\mathcal{EL}$-TBoxes $\mathcal{T}_1$, $\mathcal{T}_2$ and signature Σ

Output: **true** or **false**

for every concept name $X \in \mathsf{sig}(\mathcal{T}_1) \cup \Sigma$ **do**
  $\mathcal{B} := \mathrm{B}^{\Sigma}_{\mathcal{T}_1}(X)$
  $\mathcal{F} := \mathrm{F}^{\Sigma}_{\mathcal{T}_1}(X)$
  if $X \in \Sigma$ **then**
    if $\mathcal{T}_2 \not\models X \sqsubseteq \mathcal{F}$ **or** $\mathcal{T}_2 \not\models \mathcal{B} \sqsubseteq X$ **then**
      return true
    end if
  else if $\mathcal{T}_2 \not\models \mathcal{B} \sqsubseteq \mathcal{F}$ **then**
    return true
  end if
end for
return false

We note that the lower bound for the running time of Procedure 1 may also be exponential as the underlying problem of deciding the logical difference of two $\mathcal{EL}$-TBoxes is ExpTime-complete [6, 7].

Example 7. Continue Example 3, where $\mathsf{sig}(\mathcal{T}_1) \cup \Sigma = \{A, X, B_1, B_2, r\}$. For A, $\mathcal{B} = \mathrm{B}^{\Sigma}_{\mathcal{T}_1}(A) = A$, and $\mathcal{F} = \mathrm{F}^{\Sigma}_{\mathcal{T}_1}(A) = B_1 \sqcap B_2$. As $\mathcal{T}_2 \models A \sqsubseteq \mathcal{F}$ and $\mathcal{T}_2 \models \mathcal{B} \sqsubseteq A$, the loop continues. Then for X, $\mathcal{B} = \mathrm{B}^{\Sigma}_{\mathcal{T}_1}(X) = \mu x.(A \sqcup \exists r.x)$ and $\mathcal{F} = \mathrm{F}^{\Sigma}_{\mathcal{T}_1}(X) = B_1 \sqcap B_2$ (cf. Example 6). Since it holds that $\mathcal{T}_2 \models \mathcal{B} \sqsubseteq \mathcal{F}$, the loop continues. For B_1, $\mathcal{B} = B_1 \sqcup A \sqcup \exists r.\mu x.(A \sqcup \exists r.x)$ and $\mathcal{F} = B_1$. As $\mathcal{T}_2 \models B_1 \sqsubseteq \mathcal{F}$ and $\mathcal{T}_2 \models \mathcal{B} \sqsubseteq B_1$, the loop continues. The case of B_2 is similar to that of B_1. Finally, the algorithm returns false.

Procedure 1 can be modified to obtain witnesses to difference subsumption.

Example 8. We run Procedure 1 on $\mathcal{T}_1$, $\mathcal{T}_3 = \mathcal{T}_2 \setminus \{Z_1 \sqsubseteq B_1\}$ and Σ, where $\mathcal{T}_1$, $\mathcal{T}_2$ and Σ are the same as in Example 3. Then, for X, we have that $\mathcal{B} = \mathrm{B}^{\Sigma}_{\mathcal{T}_1}(X) = \mu x.(A \sqcup \exists r.x)$ and $\mathcal{F} = \mathrm{F}^{\Sigma}_{\mathcal{T}_1}(X) = B_1 \sqcap B_2$. However, $\mathcal{T}_3 \not\models \mathcal{B} \sqsubseteq \mathcal{F}$, which means $X \in \mathsf{cWtn}^{\mathsf{mid}}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_3)$. Similarly, it can readily be seen that $B_1 \in \mathsf{cWtn}^{\mathsf{rhs}}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_3)$.

5 Conclusion

We have revisited our solution to logical difference problem for $\mathcal{EL}$-terminologies which was based on finding simulations between hypergraph representations of the terminologies [4]. We have shown that there is a new type of witness in the logical difference between two $\mathcal{EL}$-TBoxes. We have shown that deciding the logical difference between two $\mathcal{EL}$-TBoxes can be reduced to fixpoint reasoning w.r.t. TBoxes.

References

1. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05. Morgan-Kaufmann Publishers, Edinburgh, UK (2005)
2. 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, 2 edn. (June 2010)
3. Brandt, S.: Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and—what else? In: de Mantáras, R.L., L. Saitta (eds.) Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-2004). pp. 298–302. IOS Press (2004)
4. Ecke, A., Ludwig, M., Walther, D.: The concept difference for $\mathcal{EL}$-terminologies using hypergraphs. In: Proceedings of DChanges'13. CEUR-WS.org (2013)
5. Hofmann, M.: Proof-theoretic approach to description-logic. In: Proceedings of the 20th Annual IEEE Symposium on Logic in Computer Science. pp. 229–237. LICS '05, IEEE Computer Society, Washington, DC, USA (2005)
6. Konev, B., Ludwig, M., Walther, D., Wolter, F.: The logical difference for the lightweight description logic $\mathcal{EL}$. Journal of Artificial Intelligence Research (JAIR) 44, 633–708 (2012)
7. Konev, B., Walther, D., Wolter, F.: The logical difference problem for description logic terminologies. In: Proceedings of IJCAR'08: the 4th international joint conference on Automated Reasoning. pp. 259–274. Springer-Verlag, Berlin, Heidelberg (2008), `http://dx.doi.org/10.1007/978-3-540-71070-7_21`
8. Ludwig, M., Walther, D.: The logical difference for elhr-terminologies using hypergraphs. In: Schaub, T., Friedrich, G., O'Sullivan, B. (eds.) Proceedings of the 21st European Conference on Artifical Intelligence (ECAI 2014). Frontiers in Artificial Intelligence and Applications, vol. 263, pp. 555–560. IOS Press (2014)
9. Nikitina, N., Rudolph, S.: Expexpexplosion: Uniform interpolation in general EL terminologies. In: ECAI. pp. 618–623 (2012)
10. Sattler, U., Vardi, M.Y.: The hybrid μ-calculus. In: Proceedings of IJCAR'01: the First International Joint Conference on Automated Reasoning. Lecture Notes in Computer Science, vol. 2083, pp. 76–91. Springer (2001), `http://dx.doi.org/10.1007/3-540-45744-5_7`

TourPack: Packaging and Disseminating Touristic Services with Linked Data and Semantics

Anna Fensel, Elias Kärle, Ioan Toma

University of Innsbruck, Semantic Technology Institute (STI) Innsbruck
Technikerstr. 21a, 6020 Innsbruck, Austria
{anna.fensel, elias.kaerle, ioan.toma}@sti2.at

Abstract. While the touristic service offers become present and bookable in abundance on the ICT communication channels, TourPack aims to build a linked data -empowered system for touristic service packaging. Integrating information from multiple sources and systems employing linked data as a global information integration platform, and mining from the depths of the "closed" data, the touristic service package production system will be able to cater to creating the most optimal travel experience for the traveler. Further, the service packages will be efficiently published and made bookable to the end consumers via intelligently selected most suitable communication and booking channels: especially the ICT channels with rapidly growing user audiences, such as the social media and the mobile apps.

Keywords: Services, eTourism, Semantic Technology, Linked Data, Online Booking Systems, Social Media, Mobile Apps.

1. Introduction

During the past decades, the internet, especially the Web, as well as the mobile channels, have become the most important sources for planning and booking of trips, holidays and business travels. With this trend of the Web systems gaining more importance for the touristic service ecosystem, the appearance of the Linked Open Data, in combination with conventionally used proprietary semi-structured information sources and on-line services, delivers growing and significant potential to efficiently publish and access touristic offers.

Data centric information channels currently provide machine-processable information such as mountain bike routes or public transport schedules, restaurants with specific food preferences; on-line services allow booking of hotels, skipasses, concert tickets, etc. Making touristic services easy to publish for the service providers and easy to find and book for the tourists are the key challenges for the production of a complete online service tourist offer package. The abundance and variety of travel services and the restricted time the travelers typically have on vacations or in business trips, touristic service search, selection and combination requires a lot of effort from the service consumer, when aiming at an optimal travel experience. Also, as in many service-oriented businesses nowadays, the touristic service consumers want individualized experiences and no longer want the "one-size fits all" touristic packages, as, for example, produced in a generic way by travel agencies. Thus, the aim of TourPack[1] is to design and a prototype a production system that creates "on-demand" touristic packages catering to the individual touristic service consumer needs and preferences – applying the smart usage of the open and proprietary data for the information integration and service composition, and eventually, improving the multi-stakeholder data-driven production processes of touristic service offer. Further, the pilot service prototypes will showcase the TourPack approach and infrastructure, and involve the real end user communities with varying sociodemographic factors and gender characteristics.

This paper is structured as follows. In Section 2, the current state of the art and the needed steps beyond are presented. The addressed problems and typical user scenario examples are in Section 3. The TourPack approach to the solution is described in Section 4. Finally, Section 5 concludes and summarizes the paper.

2. State of the Art and Current Knowledge

Progress beyond the state of the art in TourPack is mainly ***in more efficient and interoperable booking of travel services by delivering a technical touristic service production system solution that integrates: (1) booking through various heterogeneous channels and devices, providing mobile access not as a separate solution but as an integrated aspect of a multi-channel communication, interaction and value exchange framework, (2) service combinations of core and external added value services and (3) yield management over heterogeneous channels and devices***. The progress in these relevant main directions is as follows:

(1) Mobile channels In this world of constant connectivity, consumer interactions with enterprises have transcended to the online world. The increasing number of mobile users around the world creates new opportunities for enterprises. Nevertheless, mobile users have different expectations in the way they access the information and services that shall not be neglected, especially by the tourism business, where a bad impression on a customer might bring fatal consequences. They want to connect to enterprises wherever, whenever, however they want, and will easily move elsewhere if dissatisfied.

Furthermore, consumers are more and more interested in communication via different (and multiple) channels. The ability to answer customer demands wherever they are, and using the channel and device of their choice, will make a huge impact in their experience and consequently in the business. The fact that customers want access to all the services (Gaffney, 2007) creates the necessity of an integrated strategy. Mobile services must be integrated in the business process, not seen as a separate endeavor.

To demonstrate the importance of the mobile experience, Google (Google, 2012) took a deeper look at users' expectations and reactions towards their site experiences on mobile devices. Most interestingly, 61% of people said that they would quickly move onto another site if they did not find what they were looking for right away on a mobile site. The bottom line: Without a mobile-friendly site (that could be extended

[1] TourPack project: http://tourpack.sti2.at

to mobile access to services) one will be driving users to the competitors. Having a great mobile site is no longer just about making a few more sales. It's become a critical component of building strong brands, nurturing lasting customer relationships, and making mobile work.

Regarding the mobile experience, many customers prefer interactions via online channels rather than face to face, a fact that is currently supported by the increased number of mobile devices within the customer's reach. An appropriate mobile strategy integrated in the online multichannel world will also benefit the management and customer service for the tourism business (Revinate, 2012).

The initiative GoMo[2] from Google is a good example of best practices for companies to embrace the mobile world, providing also technical advice on how to make this adaptation while taking into account the expected effect.

In terms of current alternatives, there is a lack of integrated mobile support for a multi-channel communication and booking. In *TourPack* we enable mobility as an integrated feature, facilitating as a final goal the value exchange with the customers.

(2) Service Integration There are several approaches for spontaneous service integration: A technique to integrate web services into Jini service applications on the fly is proposed in (Gannod et al., 2003). Jini (also called Apache River) is a framework for the creation of distributed systems by the integration of modular services. In (Gannod et al., 2003) web services are used as Jini services by wrapping WDSL to Jini. The wrapping tool generates the services source code, the interface source and the Jini connection source. Using this approach the services can only be integrated into Jini applications. The solution supports the integration of WSDL based web services, but other web services like semantic web services or deep web services are not considered.

In (Leong et al., 2009) an intelligent web services architecture framework for on the fly service integration is introduced. The framework includes functionalities like service discovery, service engagement and service on the fly integration. The framework handles OWL-S, WSBPEL, WSDL, WSMO, WS-CDL and other SOA standards. However, it is not able to discover and handle deep web services.

MySIM is a spontaneous service integration middleware presented in (Ibrahim et al., 2009). It consists of four modules for the integration of services on the fly. The translator module translates all kinds of different services like web services into a generic service model. The generator module composes adequate services. The evaluation module evaluates the previous equivalence and composition relations. The last module, the builder module, implements the compositions and integrates the services in a chosen technology model. MySIM offers techniques for the spontaneous integration (translation) of OSGi and standard web services, but it cannot translate and integrate services from the deep web.

Our approach to the service integration will close the gaps of the approaches introduced above and create a production solution for the dynamic integration of touristic web services and deep web services on the fly that will enable the creation of enhanced integrated services that combine core and external added value services.

---

[2] http://www.howtogomo.com

(3) Yield management Yield management[3] (Weiß, S. Haüßler, 2005) refers to the business activities that companies are doing in the scope of maximizing profits from a fixed and finite resource (e.g. availability of lodging businesses, airplane seats, etc.). Yield management could be considered as a multi-disciplinary concept as it needs information and data from various sources and departments. In this world of multi-channel distribution, multi-channel booking and multi-channel communication, the application of yield management in the most effective way is becoming a tough challenge for researchers to solve and make it available for use to the various business domains. The heterogeneity of the multi-channel ecosystem hinders the maintenance of offers as there are different constraints introduced from the various channels. Moreover, the major objective of maximizing the profits from a limited resource should be extended to cover the offering of combined services from various service providers.

Tourism is a domain with many cases relevant to the objectives of yield management (Amersdorffer et al., 2010). In the hotel industry it is also known as revenue management (Fandel, 2005). For instance, an hotelier has a finite number of rooms, which should be sold in a way that the profit is maximized and the cost is reduced to a minimum level. Yield management in tourism consists of various aspects like capacity management (Xylander, 2003), overbooking, dynamic pricing, length of stay (e.g. a lot of hotels are promoting offers with the two nights stays minimum due to yield management results), price limits (i.e. in relation to the average rate in the city of the hotel), last-minute reservations etc. In this respect hoteliers need to employ the appropriate tools in order to properly apply yield management and monetize its benefits. An example of such a tool is the Amadeus Hotel Platform[4], which helps hoteliers to follow the revenue management objectives and fill their hotel rooms or other service capacities at the most profitable price.

According to (Hayes and Miller, 2010) the revenue management lifecycle for hotels includes five major steps, namely: a) establish prices, b) forecast demand, c) manage inventory, d) manage distribution, and e) evaluate results. The first step consists of the price establishment of the offered services and incorporates feedback from the last step of the previous iteration (in case it exists), the evaluation of results. Afterwards, the customer demand can be estimated ("forecast demand"), the management of the available rooms ("manage inventory") is required and the distribution channels should be carefully managed to maximize revenue. The management of the distribution channels should be done in a way that minimizes the transaction costs and supports the maximization of the profit. Furthermore, the lifecycle of the yield management needs to be adapted to the offer of combined services in order to cover the package offerings that tourism businesses are promoting in nowadays (e.g. accommodation package integrated with car rental services).

The aim of the proposed solution is closely related to the steps that are considered crucial for the materialization of yield management. We enable the touristic service provider to manage the multi-channel communication and incorporate feedback that is gathered through the "forecast demand" and the "evaluate results" phases, which are the second and fifth steps of revenue management, re-

---

[3] http://en.wikipedia.org/wiki/Yield_management

[4] http://www.amadeus.com/hotelit/hotel-platform.html

spectively. In addition, the optimal management of the distribution channels (i.e. "manage distribution" phase) will be facilitated by minimizing the transaction costs and maximizing the profit via the direct bookability of the tourism services.

3. Description of Problem and User Scenarios

The internet, web-based communication and booking channels are becoming increasingly important in today's completive world. Organizations of all sizes, commercial and not-for-profit, regularly face the challenge of communicating with their stakeholders using a multiplicity of channels, e.g. websites, videos, PR activities, events, email, forums, online presentations, social media, mobile applications, and recently structured data.

The social media revolution has made this job for the organisations – as well as for their customers when spending time on learning about service offers - much more complicated, because:

- the *number of channels* has grown exponentially,
- the communication has changed from a mostly unilateral "push" mode (one speaker, many listeners) to an increasingly fully *bilateral communication*, where individual stakeholders (e.g. customers) expect one-to-one communication with the organization, and the expected speed of reaction is shrunk to almost real-time, and
- the *contents of communication is becoming increasingly granular* and more dependent upon the identity of the receiver and the context of the communication.

On the other hand, the booking market is moving online. In this context, data centered platforms – e.g. supporting booking, social media and mobile presence, are also becoming new dissemination and even main channels for **touristic service providers** to reach the customers. Currently, there are more than 100 booking platforms available on which the hotelier could be present.

Hence, the first challenge that needs to be addressed is ***visibility.*** To be found by relevant customers the tourism service provider needs to ensure to be have the outreach to the most relevant customers as possible. This requires apart from time and resources, competence in the field of online marketing and commerce (which is intended to be supported by the interoperable intelligent service composition mechanisms).

This highlights the challenge of ***scalability***, which is another problem that needs to be addressed in this context. The average time required for a service business to maintain a profile of a medium sized hotel at one portal is between 5 to 15 minutes a day. An effort of maintaining a business's profile on 100 portals would then require at least 20 hours of work which for a medium size business, is a lot of time, effort and finally money that has to be invested in something that distracts the hoteliers from focusing on the core business. Tourism service providers are thus facing a challenging multi-channel problem by having to maintain the right balance of rooms' availability across more than 100 channels on a daily basis. This obviously does not scale. Being accurate and visible in all the channels is a must in order to increase revenues. Yield management also plays an important role in this context. Adopting your offer and your price dynamically in response to the behavior of your (on-line visible) environment and selecting the right channel and customer will become critical to economic success.

In addition to the growing number of online channels, there are an increasing number of possibilities to access them. Mobile devices have become a popular means to access and book tourist related content and services online. It is therefore crucial for hotels to also be bookable through mobile devices, since most of the bookings will be done via mobile applications in future.

Due to these recent developments, competence in on-line communication and marketing as well as on-line sales is ***crucial for ensuring the competiveness of a country with a large tourism intake***. Losing the value of bookings via payment of commissions should be limited as far as possible. Consider the booking of hotel rooms as an example. More than 12% of all on-line hotel room bookings in Austria are done through hotel booking channels such as HRS or booking.com[5]. A portal such as HRS takes 18% of the price of a hotel room for offering this service. On a global scale, we are also seeing rising worldwide competitors such as Google, that are defining and implementing new business models and techniques for online marketing and booking that may once again change the transfer and distribution of these fees. Losing control and competence in this cornerstone of the tourist value chain may generate significant risk for the economic and social future of Austria as a touristic destination. Maintaining competence and competiveness in on-line marketing may therefore be key to future prosperity.

Summing up it can be said that touristic service providers need an integrated production solution that provides management and execution of communication and booking goals primarily in an automated fashion, with costs equivalent to mass-media communication, along with the granularity of individual experts, and at the pace of real-time social media. We are aiming to mechanize important aspects of these tasks, allowing scalable, cost-sensitive, and effective communication for small-or-medium sized business units and comparable organizations for which information dissemination is essential, but resources are significantly limited. Considering these challenges, it is crucial for all touristic service providers to introduce appropriate technical solutions to be competitive in a future online world and to maintain their current ability to participate in the economic tourist value chain.

On the other hand, for **touristic service consumers**, the data-driven production service system would certainly be crucial when finding and consuming the most relevant services on the fly. Below, as typical end user scenarios, two short stories describe a customer (guest) on an average day on holidays and show how the software can be integrated in a hotel's and a hotel-guest's every-day business.

End user scenario A. A guest G enters the hotel for the first time. At the check-in desk the receptionist introduces G to the newly launched smartphone app of the hotel. G downloads the app in the free WiFi of the hotel and back in his/her room he/she starts exploring the contents. In the "restaurants"-section of the app she/he finds the menus of the day generated on the fly from linked data of the available restaurants in the nearby, catering to the user’s food preferences and dietary restrictions. Since she/he feels quite hungry he/she makes a reservation for a certain preferred type of restaurant in the area directly out of the app (Fig. 1).

[5] http://www.slideshare.net/Roli1219/the-power-of-online-traval-agencies-ota (slide 25)

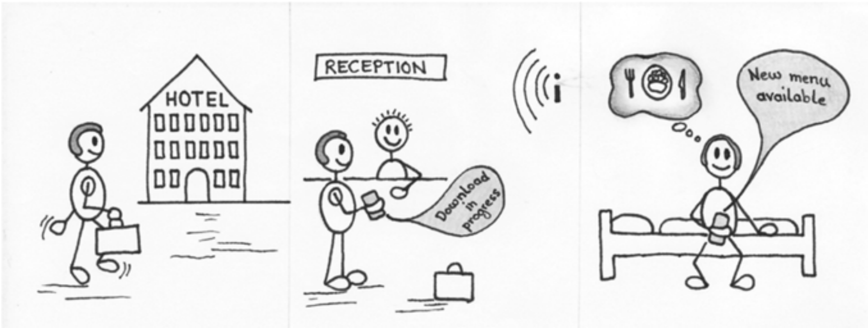


Fig. 1: Guest visits hotel and installs app (Illustration: Caroline Winklmair)

End user scenario B. After G made her/his reservation she/he struggles to find the restaurant the reservation is made at. The tourist service consumer takes out his/her phone, browses to the page of the touristic service production data and finds the preferred transportation directions – based on the open data of public transport, taxi services and maps, which easily guide him/her to the desired lunch (Fig. 2).

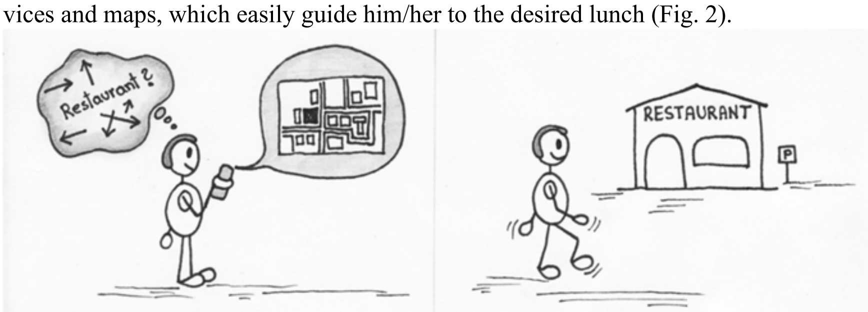


Fig. 2: Guest navigates to restaurant with the linked data empowered map (Illustration: Caroline Winklmair)

Naturally, the touristic service production system would extend to heterogeneous types of services (e.g. in wellness, shopping, sports, culture), and in the same personalized manner would deliver to the end users packaged offers for experiences matching their expectations.

4. Goals and Results

TourPack generates the next generation of technologies for eTourism that can be easily deployed in the hospitality industry and needed to ensure visibility, interaction, and access to tourist services. Specifically, TourPack aims at providing pragmatic technology for online touristic service offer production and its efficient and scalable multi-channel online communication and booking through a multitude of channels (i.e. web sites, wikis, social media channels) through various mobile devices. The major TourPack technical objectives are:

i. ***design and implement a scalable online service packaging and provisioning solution based on machine-processable semantics.*** Scalability is achieved by introducing a layer of abstraction over all communication channels as well as a layer for capturing customer domain information. These two layers can then be dynamically mapped and connected, depending on the particular use case and direction of information propagation (publishing of messages or collection of feedback).
ii. ***deliver the technology for interacting with this multi-channel solution through various and heterogeneous mobile channels.*** We will develop a mobile toolkit that can be used to develop adapters, integrate and use mobile channels into the *TourPack framework*.
iii. ***provide support in service packaging, such as accessing, interacting, and value exchange (i.e., booking)*** of tourism services and their combinations through this infrastructure, using linked data as a global integration platform. We would like to support the hospitality industry in optimizing their revenue and profit management through easy and liquid booking in numerous channels and through numerous devices. We provide support in empowering the service provider towards low-fee (e.g., direct) booking opportunities to reduce the share of the income that is taken by external booking providers.
iv. ***validate and apply the TourPack research and development outcome in pilots focusing on the booking of tourism services.*** We will show how *TourPack* technology will enable tourism enterprises to simplify and automate their communication activities, to engage possible customers via this multitude of channels, to gain visibility and in the end to increase their income by gaining more direct booking but also by saving on fees that are required by some of the booking channels.

Today's service ecosystems, including the ones addressing travel service offers, deal with increasing quantities of unstructured and semi-structured information in e-mails, text documents, spreadsheets, webpages, news articles, collaborative posts, social media to name but a few. Unstructured and semi-structured information is a vital part of an enterprise, for daily operations as well as for long-term strategic management. While these resources contain truly valuable contents, they are of limited use if they cannot automatically be handled by applications. Extracting knowledge from unstructured and semi-structured sources is the focus of Information Extraction (IE) research. TourPack developments are aiming at providing a state-of-the-art IE-driven semantic tool for (semi-)automatic touristic service semantic annotations and packaging in order to heavily reduce manual data entry for the population of the touristic service providers and consumers. IE is often supported by domain ontologies in order to identify the ontological concepts and relations that semantically describe the text content (Kiryakov et al., 2003). Proof of concept systems such as SOFIE (Suchanek et al., 2009) can parse natural language documents and extract ontological facts. Systems such as YAGO (mpi-inf.mpg.de/yago-naga) and Kylin/KOG (Wu and Weld, 2008) exploit supplementary semantics from Wikipedia and WordNet to extract semantically enhanced information from textual data. Semantic annotation platforms such as KIM (ontotext.com/kim), GATE (http://gate.ac.uk) or OpenCalais (http://www.opencalais.com) locate and extract entities, relationships, and facts in texts, and create semantic links between different documents, data, domain models, and Linked Data. Once extracted from various sources, relevant manufacturing ecosystem knowledge can be inter-linked and then clustered in order to enable better search and navigation of virtual artefacts. This can be realized using approaches and techniques such as those provided by LarKC (larkc.eu) or LOD2 (lod2.eu). LarKC developed methods and tools to recognize entities and relations, and to interlink these entities with existing documents to provide richer search experiences. LOD2 supported Sindice (sindice.com), a platform for building applications on top of RDF-based Linked Data.

We will support automatic generation, clustering and packaging of semantically annotated touristic service offers from a variety of sources. More precisely, existing information extraction, clustering and publishing will be adopted and extended in order to:

- obtain the extracted data in a Linked Data format, (semi-)automatically associating metadata;
- generate service representations in Linked Data format according to ontological models;
- interlink, cluster, package and provide services in an automatic way;
- provide a semantic service and an online interface for easy publishing and access to the above mentioned functionalities.

As confirmed by the industry partners involved in this proposal, costs remain the main decisive factor for SME adoption of innovative ICT solutions. We plan to build upon open source tools for information extraction, interlinking and clustering many of the ones mentioned above. In particular we will consider OpenCalais for information extraction as well as linking and clustering tools developed by LarKC and LOD2 projects.

Further, the TourPack approach addresses a number of innovative challenges, and expected technical outcome and own contributions of the project are summarized in the following table:

Challenge	Outcome
Multi-channel communication see (Fensel et al., 2012), (Toma et al., 2013), (Fensel et al., 2014)	• Semantic based representation of content (ontology) in intuitive and familiar terminology for tourist service providers. • Scalable methods for separating and interweaving content and communication channels, particularly, employing linked data as an integration platform. • Online multi-channel communication technical solution.
Online interactions see (Fensel et al., 2014), (Stavrakantonakis et al., 2014-1)	• Formal communication pattern description mechanism as business processes. • Reusable set of communication patterns to structure the online interactions for the tourism domain.

Service integration and yield management see (Toma et al., 2014), (Stavrakantonakis et al., 2014-1), (Stavrakantonakis et al., 2014-2)	• Integration of a booking engine with the necessary infrastructure for tourism services to be directly bookable and configurable for yield management and tailored to the preferences of the end consumers. • A technique for enablement touristic service providers to annotate their offers employing linked data for the subsequent multi-platform reuse.
Mobile service provisioning see (Kärle, 2014), (Davies et al., 2011), (Qiao et al., 2015)	• Online mobile strategy definition for tourism organizations. • Mobile toolbox for the integration of booking services for travel service providers. • Mobile framework and components for multi-channel and online interactions management.

6 Conclusion

This paper presents the TourPack approach to designing, developing and deploying touristic service packages based on semantic technology as an enabler for tourists and tourism businesses to participate productively by providing new experiences and finding new direct dissemination and booking channels, while leveraging on touristic data value chain.

The effort already runs pilots, such as with Touristic Association of Innsbruck, and already implemented semantic dissemination support by implementing schema.org support on their website[6] and publishing the touristic data of the Innsbruck region as linked open data (Toma et al., 2014). Also, in cooperation with SalzburgerLand, the touristic data of Salzburg are published in Linked Open Data format with schema.org, and are usable[7]. We are deploying our solution also with direct touristic service providers: starting with hotels, and extending to further touristic services.

Acknowledgments. This work has been partially funded by projects TourPack and OpenFridge, supported by the Austrian Research Promotion Agency (FFG) within the program "Future ICT", FWF-funded project OntoHealth, as well as an EU-funded project BYTE, and ÖAD project RESIDE. The author gratefully acknowledges useful inputs of TourPack industry partners: Austrian SMEs m-Pulso, Redlink and Seekda, members of OC working group[8], as well as the STI Innsbruck's start-up ONLIM[9], specializing on online social media marketing.

[6] Website of Touristic Association of Innsbruck: www.innsbruck.info
[7] SalzburgerLand Data Hub: http://data.salzburgerland.com
[8] Online Communication (OC) Working Group: http://oc.sti2.at
[9] ONLIM – Online Communication and Marketing Tool: http://onlim.com

References

Revinate. (2012). Using Social Media and Mobile to Improve Guest Satisfaction - The Revinate Blog. Retrieved from http://blog.revinate.com/2012/09/using-social-media-and-mobile-to-improve-guest-satisfaction.html

Google (2012). “What Users Want Most From Mobile Sites Today”. Google survey conducted by Sterling Research and Smith Geiger, July 2012. Retrieved from: http://www.thinkwithgoogle.com/insights/library/studies/what-users-want-most-from-mobile-sites-today/

Gaffney, A.: Improving Customer Satisfaction Through More Effective ICT Strategies. Vol. 1, pp. 1–7, 2007.

Hayes, D.K., and Miller, A.A.: Revenue Management for the Hospitality Industry. Book, Wiley 2010.

Xylander, J.K.: Kapazitätsmanagement Bei Reiseveranstaltern: Entscheidungsmodelle Zur Kontingentierung Im Yield Management. Deutscher Universitätsverlag, 2003.

Fandel, G.: Revenue Management. ZfB Special Issue 1/2005, Gabler.

Amersdorffer, D., Bauhuber, F., Egger, R., and Oellrich, J.: Social Web im Tourismus. Springer, 2010.

Weiß, M., Haüßler, S.: Yield Management – Konzepte und Techniken. Grin, 2005.

Gannod, G.C., Zhu, H. and Mudiam, S.V.: On-the-fly Wrapping of Web Services to Support Dynamic Integration. In Proceedings of the 10th Working Conference on Reverse Engineering (WCRE '03). IEEE Computer Society, Washington, DC, USA, 175-184, 2003.

Leong, H.W., Brutzman, D., McGregor, D. and Blais, C.: Web services integration on the fly for service-oriented computing and simulation. In Proceedings of the 2009 Spring Simulation Multiconference (SpringSim '09). Society for Computer Simulation International, San Diego, CA, USA, Article 40 , 8 pages, 2009.

N. Ibrahim, F. Le Mouël, and S. Frénot: MySIM: a spontaneous service integration middleware for pervasive environments. In Proceedings of the 2009 international conference on Pervasive services (ICPS '09). ACM, New York, NY, USA, 1-10, 2009.

Kiryakov, A., Popov, B., Terziev, I., Manov, D. and Ognyanoff, D. (2003) Semantic Annotation, Indexing, and Retrieval, In Proceedings of the 2nd International Semantic Web Conference (ISWC 2003), pp.49–79.

Suchanek, F. M., Sozio, M. and Weikum, G. (2009), SOFIE: A Self-Organizing Framework for Information Extraction, in Proceedings of WWW 2009, Madrid.

Wu, F., and Weld, D. (2008) Automatically refining the Wikipedia infobox ontology, in Proc. of WWW 2008, Beijing.

Fensel, A., Fensel, D., Leiter, B., Thalhammer, A. " Effective and Efficient Online Communication: The Channel Model". In Proceedings of International Conference on Data Technologies and Applications (DATA’12), pp. 209-215, SciTePress, Rome, Italy, 25-27 July, 2012.

Toma, I., Fensel, D., Gagiu, A.-E., Stavrakantonakis, I., Fensel, A., Leiter, B., Thalhammer, A., Larizgoitia, I., Garcia, J.-M. "Enabling Scalable Multi-Channel Communication through Semantic Technologies". In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, November 17-20, 2013, Atlanta, USA, IEEE Computer Society Press (2013).

Toma, I., Stanciu, C.-V., Fensel, A., Stavrakantonakis, I., Fensel, D. "Improving the online visibility of touristic service providers by using semantic annotations". In Proceedings of the 11th European Semantic Web Conference (ESWC'2014) – Poster and Demo Track, 25-29 May 2014, Crete, Greece, Springer-Verlag, LNCS (2014).

Stavrakantonakis, I., Fensel, A., Fensel, D. "Matching Web Entities with Potential Actions", In Proceedings of the Poster Paper track International Conference on Semantic Systems (I-SEMANTICS’14), CEUR-WS Vol. 1224, 4-5 September 2014, Leipzig, Germany (2014).

Stavrakantonakis, I., Toma, I., Fensel, A., Fensel, D. “Hotel Websites, Web 2.0, Web 3.0 and Online Direct Marketing: The Case of Austria”. In Proceedings of the 21th International

Conference on Information and Communication Technologies in Travel and Tourism (ENTER2014), pp. 665-677, 21-24 January 2014, Dublin, Ireland, Springer (2014).

Kärle, E. Master thesis "An approach and implementation for supporting multi-platform mobile service creation for businesses", January 2014, University of Innsbruck, Austria.

Davies, M., Carrez, F., Heinilä, J., Fensel, A., Narganes, M., Danado, J. "m:Ciudad -- Enabling End-User Mobile Service Creation", International Journal of Pervasive Computing and Communications, Emerald Group Publishing, Vol. 7 Iss: 4, pp. 384-414 (2011).

Qiao, X., Xue, S., Chen, J., Fensel, A. "A Lightweight Convergent Personal Mobile Service Delivery Approach Based on Phone Book". International Journal of Communication Systems, Vol. 28 Iss: 1, pp. 49-70, Wiley, January 2015.

A Practical Framework for *RelBAC* Implementation

Lei Liu[1], Quanzhu Tao[1], Fausto Giunchiglia[2] and Rui Zhang[1]⋆

[1] CCST, Jilin University, Str. Qianjin 2699, Changchun, China 130012
liulei@jlu.edu.cn
tqzlimit@126.com
rui@jlu.edu.cn
[2] DISI, University of Trento, Italy
fausto@disi.unitn.it

Abstract. *RelBAC* is a new access control model that has gradually aroused the research interest in the domain of access control. But it is still not mature enough for industrial application due to its high logical complexity. In this paper, we present a framework to implement *RelBAC*. First, access control queries to *RelBAC* knowledge base (KB) are analysed and categorized into different queries as run-time or off-line. Then the necessary knowledge is studied to answer each type of query. We propose to separate the knowledge for run-time query, named as a complete ABox, from the classical *RelBAC* KB and store it in a relational database, so as to provide run-time answers within acceptable time. Last, a theorem is proved to backbone our method and an algorithm is proposed to calculate the complete ABox. This framework serves as a meaningful attempt to put *RelBAC* into practice.

Keywords: *RelBAC*, Description Logic, Complete ABox

1 Introduction

Access control models evolute with the advances of technologies. The famous Role-based Access Control model RBAC [4] was proposed in the 20th century, has now flourished in practical access control systems such as in Windows operation systems. It evolves into different models such as ARBAC [11], GeoRBAC [3], ARBAC [8] etc.

As a new access control model, *RelBAC* [6] connects the subject that query to access the object with the permission as a binary relation. The model provides intuitive and straightforward semantics to home users without formal access control experiences. But *RelBAC* has not flourished as expected in industry solutions. What hinders the model from practical application is the shortage of background supporting mechanism, which is supposed to be the powerful reasoning services provided by description logic (DL) reasoners. There is no good enough general purpose reasoners for *RelBAC* yet.

⋆ Corresponding Author

In this paper, we propose a practical framework to implement *RelBAC*. The knowledge base (KB) of a *RelBAC* access control system can be classified into different parts, namely, organization, authorization, constraint and environment. Each part consists of its typical structured axioms or assertions. Such structures are studied and classified into run-time and off-line queries to the KB. We prove that to answer the run-time query, only part of the ABox assertions are necessary. An algorithm is proposed to populate individuals of concepts to build a complete ABox. Such ABox assertions are separated from the classical KB and stored directly in a relational database. Then query from end user and/or system can be parsed and distributed to different engines, i.e. the database query engine to provide run-time response and the reasoning engine to check off-line queries. The framework provides a practical path to implement *RelBAC* in industrial solutions.

The paper is organized as follows. Section 2 gives a glance in *RelBAC* model; Section 3 illustrate our framework; Section 4 shows our strategy to implement *RelBAC*; and we conclude in Section 5.

2 Preliminaries

Relation-based Access Control (*RelBAC*) is an access control model introduced in [6] and formalized using the DL $\mathcal{ALCQIBO}$ as described in [16]. In this section, we will illustrate the basic *RelBAC* definitions and related access control policies.

2.1 Elementary

As shown in the ER Diagram of Figure 1, what distinguishes *RelBAC* from other access control models is the way it models permissions. A `PERMISSION` is modeled as an operation that users (`SUBJECT`s) can perform on certain resources (`OBJECT`s). To capture this intuition, a `PERMISSION` is named with the name of the operation it refers to. The *generalization* (loops) on each component repre-

Fig. 1. ER Diagram of *RelBAC*.

sents `IS-A` relations. Not only `SUBJECT` and `OBJECT` are organized along `IS-A` hierarchies but also `PERMISSION`.

2.2 Formalization

Together with *RelBAC*, a logic $\mathcal{ALCQIBO}$ extends the DL $\mathcal{ALC}$ [2] with qualified cardinalities, inverse roles, nominals and Boolean for roles (see [12, 10, 9] for extensions of DLs with Booleans between roles). As described in [1], $\mathcal{ALCQIBO}$ is applied in access control domain to formalize *RelBAC*.

$\mathcal{ALCQIBO}$. The syntax of $\mathcal{ALCQIBO}$ is defined as follows.

$$
\begin{aligned}
C, D &::= A \mid \neg C \mid C \sqcap D \mid \geq n\, R.C \mid \{a_i\} \\
R, S &::= P \mid R^- \mid \neg R \mid R \sqcap S
\end{aligned}
$$

where $A \in \mathsf{N_C}$, $P \in \mathsf{N_R}$, $a_i \in \mathsf{N_I}$ and $n \in \mathbb{N}$.
A KB (KB) is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ where $\mathcal{T}$, called *TBox*, is a finite set of *general concept inclusions (GCIs)* of the form $C \sqsubseteq D$ and a finite set of *general role inclusions (GRIs)* of the form $R \sqsubseteq S$, while $\mathcal{A}$, called *ABox*, is a finite set of concept and role assertions of the form $C(a_i)$ and $R(a_i, a_j)$, with $a_i, a_j \in \mathsf{N_I}$.

The corresponding semantics (partial) of $\mathcal{ALCQIBO}$ is defined as follows.

$$
\begin{aligned}
(R^-)^{\mathcal{I}} &:= \{(y, x) \in \Delta \times \Delta \mid (x, y) \in R^{\mathcal{I}}\}, \\
(\neg R)^{\mathcal{I}} &:= \Delta \times \Delta \setminus R^{\mathcal{I}}, \quad (\neg C)^{\mathcal{I}} := \Delta \setminus C^{\mathcal{I}}, \\
(R \sqcap S)^{\mathcal{I}} &:= R^{\mathcal{I}} \cap S^{\mathcal{I}}, \quad (C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}, \\
(\geqslant n\ R.C)^{\mathcal{I}} &:= \{x \in \Delta \mid \sharp\{y \in \Delta \mid (x, y) \in R^{\mathcal{I}} \text{ and} \\
&\qquad\qquad y \in C^{\mathcal{I}}\} \geq n\}, \\
\{a_i\}^{\mathcal{I}} &:= \{a_i^{\mathcal{I}}\}.
\end{aligned}
$$

An $\mathcal{ALCQIBO}$-interpretation $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$ is said to be a *model* of a KB, $\mathcal{K}$, iff it satisfies $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, for all $C \sqsubseteq D \in \mathcal{K}$, $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$, for all $R \sqsubseteq S \in \mathcal{K}$, $a_i^{\mathcal{I}} \in C^{\mathcal{I}}$, for all $C(a_i) \in \mathcal{A}$, and $(a_i^{\mathcal{I}}, a_j^{\mathcal{I}}) \in R^{\mathcal{I}}$, for all $R(a_i, a_j) \in \mathcal{A}$. In this case we say that $\mathcal{K}$ is satisfiable and write $\mathcal{I} \models \mathcal{K}$. A concept C (role R) is *satisfiable w.r.t.* $\mathcal{K}$ if there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that $C^{\mathcal{I}} \neq \emptyset$ ($R^{\mathcal{I}} \neq \emptyset$).

Formal Specification. In *RelBAC* we distinguish five different kinds of specifications that, altogether, constitute an access control system: the organization information, the authorization policy, the control constraint, the environment factors and the administration query. *RelBAC* uses the description logic $\mathcal{ALCQIBO}$ to express each specification by associating a concept name to each `SUBJECT` and `OBJECT` while permissions are described by means of role names.

1. **Organization**, to organize the entities and relationships among entities into hierarchical structures with partial order.
2. **Authorization**, to declare the permissions from `SUBJECT` to `OBJECT`.
3. **Constraint**, to declare general regulations that existing and new authorization policies should follow.
4. **Query**, to check the instantiation or satisfiability of the current access control KB.

The first step of our work is to clarify the patterns latent in above specifications in order to analyze the difference of the reasoning services related to each pattern.

3 Framework

Based on the theory of *RelBAC*, we propose a framework towards the implementation of the theory. It consists of three layers coherent to a standard MVC structure, but specialized to fit the access control domain.

Figure 2 gives the conceptual model of the frame.

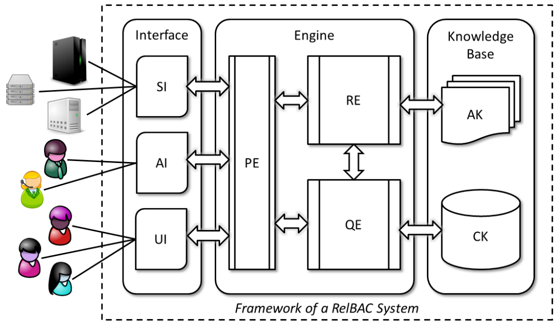


Fig. 2. Conceptual model of the frame for *RelBAC* Model.

As is shown in the figure, the framework consists of three major components, interface, engine and KB. Let us get into the details of them one by one.

Interface It is the intermediate between the system internal components and the outer 'users'. A channel, predefined or constructed at runtime, servers as interface is maintained by this component. It is not bounded to classical user interface but provides three sub components, SI, AI and UI.

- **SI** It stands for **S**ystem **I**nterface, which communicates with the outer information providers, such as time server, behavior monitor, audit record server, etc. Environment information are mainly transmitted to the system through SI.
- **AI** It stands for **A**dministrative **I**nterface, which connects administrators to the system via predefined graphical interfaces and facilitate tool interfaces.
- **UI** It stands for **U**sser **I**nterface, that provides classical graphical interfaces to end users. AI pages could be reused for UI, only with appropriate access control.

Engine It is the core part of the frame, that processes the info (updates, queries, maintenance, etc.) and communicate (if needed) with the KB. It mainly consists of three sub engines, PE, RE and QE.

- **PE** It is the **P**arser **E**ngine, which will preprocess the information from the interface, and forward it to appropriate engines for further processes. Different type of system queries will arrive, and fit in predefined or runtime studied patterns, then re-formulated into processable intermediate format, thereafter be sent to responsible engines. Details of the formats and patterns will be discussed in Section 4.

RE It is the **R**easoning **E**ngine, that accepts OWL-API format reasoning tasks and provides reasoning services together with the interaction with the ontology KB(s).

QE It is the **Q**uery **E**ngine that takes the database queries as input and provides necessary query answers via access to the database in the KB.

KB It stores the knowledge necessary for the access control system functionality, such as the organization of the `SUBJECT`, `OBJECT` and `PERMISSION`; the authorization and constraint policies; the environment factors, etc. It is divided into two parts, the AK and DK, which stores the knowledge for different purposes.

AK It is the **A**dministrative **K**nowledge base, aims at general policies that support the administrative queries. It usually interacts with RE as the reasoning background.

CK It is the **C**ontrol **K**nowledge implemented via classical relational database techniques. Instantiated knowledge, such as the ABox assertion that 'John is permitted to update the root path' is stored here.

With the framework, the next step is to work out the details of each component. We will describe the theoretical details in the next section.

4 Implementation

This section will focus on the theoretical aspects to implement the framework introduced above. The patterns are clarified for the knowledge in a *RelBAC* system. Then the distribution strategy is proposed in details.

4.1 Knowledge Pattern

One of the key issue to implement the framework is to clarify the possible knowledge patterns. In Section 2.2, knowledge has been classified into four categories. Here, we will identify the patterns from the categories.

1. **Organization**: In the access control terminology, `SUBJECT`, `OBJECT` and `PERMISSION` may be organized in a taxonomy along the `IS-A` relation [5]. An `IS-A` relation is represented as a concept (or role) *inclusion* axiom in *RelBAC*:
$$C \sqsubseteq D \text{ or } P \sqsubseteq Q \tag{1}$$
where C, D are both `SUBJECT`s or both `OBJECT`s; P, Q are both `PERMISSION`s. `SUBJECT` and `OBJECT` are easy to be organized into `IS-A` hierarchies, especially with concerned attributes. Besides, compound concepts constructed with classical DL operator $\neg$ and $\sqcap$ may also exist in the formula above. The inversion operator $^-$ on role may also join in the *inclusion* axiom of roles in Formula (1).

Moreover, in addition to the DL roles for `PERMISSION`, classical roles may also exist to describe binary relations between entities that is not a `PERMISSION`, such as 'is-older-than' or 'has-published' relation in a system with an academic background. The *inverse role* operator

The number restriction operator $\geq$ is seldom used in organization. Its combination with the other operators will results in an axiom that could not be instantiated, as it puts restrictions on cardinality rather than fix individuals. This is more likely to be used in the specification of constraints.

2. **Authorization**: It specifies a permission existing between a `SUBJECT` and an `OBJECT` both on organization level or instance level in various of forms.

$$S \sqsubseteq \forall \neg P . \neg O \text{ or } O \sqsubseteq \forall \neg P^{-} . \neg S \tag{2}$$

which specifies that any `SUBJECT` in S is permitted to perform the operation (with the same name) P on any `OBJECT` in O. For easy reading purpose, it is transformed into a SWRL [14] rule as

$$S(?x), O(?y) \rightarrow P(?x, ?y) \tag{2$'$}$$

Special cases exist for Formula (2′), with alternation of the concept(s) with nominal(s), we have the following variations.

$$\begin{aligned} \{..., s_i, ...\} &\sqsubseteq \forall \neg P . \neg O \\ S &\sqsubseteq \forall \neg P . \neg \{..., o_j, ...\} \\ \{..., s_i, ...\} &\sqsubseteq \forall \neg P . \neg \{..., o_j, ...\} \end{aligned} \tag{2$''$}$$

where i, j are natural number indexes for individuals of `SUBJECT` and `OBJECT`. An authorization policy in the form of an ABox assertion $P(s, o)$ is only a special case of Formula (2″).

3. **Constraint**: It specifies the restrictions or regulations that the authorization policies should not violate. A constraint usually stays inactive in a running system. But when environment factors change, such as the rise of the system load, the crucial time point, the access behavior violation, etc. Therefore, the reasoning engine should check the consistency of the KB when new knowledge arrives or current knowledge updates.

Some of the most concerned constraints are:

(a) **Separation of Duties (SoD)**: It regulates the mutual exclusiveness of permissions. SoD lies in different levels and granularity. Given a set of positions $\mathcal{S} = \{S_1, \ldots, S_n\}$, where each S_i is a concept name, a SoD policy 'a subject can take all the positions in $\mathcal{S}$', may take the form of an unsatisfiable compound role (in contrast to an atomic role).

$$\bigsqcup_{i=1}^{C_n^m} \left(\bigsqcap_{j=1}^{m} S_{i_j} \right) \sqsubseteq \bot \tag{3}$$

where C_n^m is the binomial coefficient of 'n choose m'. A special case of Formula (3) is in condition of $m = n$, then the formula changes into

$$\prod_{j=1}^{n} S_j \sqsubseteq \bot \tag{3'}$$

.

(b) **Chinese Wall (CnW)**: The Chinese Wall property regulates conflict of interest (CoI), that 'the resources in the set of CoI could not be accessed by the same user'. Given a set of sensitive resources, $\mathcal{O} = \{O_1, ..., O_n\}$, and the corresponding operation $\mathcal{P} = \{P_1, ..., P_n\}$, a CnW policy may take the form as.

$$\prod_{i=1}^{n} \exists P_i.O_i \sqsubseteq \bot \tag{4}$$

Specifically, when the operation remains the same, say P, then the formula changes into

$$\geq 2\ P.(\bigsqcup_{i=1}^{n} O_i) \sqsubseteq \bot \tag{4'}$$

4. **Query**: A query searches for subjects, permissions and/or objects from the KB. A consistent KB is denoted as Σ hereafter. A query is classified as either control or administrative.

Control Query (CQ) It verifies wether a requester for the resource have or not the permission to access certain number of objects. Given a `SUBJECT` u, the query could be of the following patterns.

(a) whether a `SUBJECT` s has `PERMISSION` P on the `OBJECT` o;
(b) whether a `SUBJECT` s has `PERMISSION` P on the `OBJECT` in O.

correspond to the, so called, *instance checking* reasoning service:

$$\Sigma \models P(s, o), \tag{5}$$
$$\Sigma \models (\forall \neg P.\neg O)(s). \tag{6}$$

in which Formula (6) could be reformed into a satisfiability check as in Formula (8) and (9). We will discuss it later at the end of this subsection. It is obvious that all these three formulae of CQ should be answered within acceptable time by the system.

Administrative Query (AQ) It checks the state of the access control system, such as

(a) search for all the `SUBJECT` that has `PERMISSION` P on `OBJECT` o;
(b) search for all the `OBJECT` that is permitted to some `SUBJECT` s via `PERMISSION` P;
(c) whether the current system KB is consistent;
(d) whether an intended policy implied in the KB;
(e) whether an intended policy conflicts with the KB;
(f) whether an intended policy irrelevant to the KB;

(g) whether an intended update of environment violates the KB;
(h) whether an intended update consistent with the KB;
(i) whether an intended update irrelevant to the KB;

The first two AQ's lie in the reasoning of *instance retrieval* for a compound concept, say $\exists P.\{o\}$ and $\exists P^{-}.\{u\}$, as the following,

$$\text{retrieve all } \mathtt{SUBJECT} \text{ u that} \Sigma \models \exists P.\{o\}(u) \tag{7}$$

$$\text{retrieve all } \mathtt{OBJECT} \text{ o that} \Sigma \models \exists P^{-}.\{u\}(o) \tag{8}$$

Here the compound concept is not bounded to permission, environment attributes could be considered too. Therefore the concept on the right side of $\models$ could be *conjunction* or *union* of multiple concepts. The third AQ lies in *consistency checking* for the KB.

$$\Sigma \models \bot \tag{9}$$

If there is no model exists for Σ, the answer is *inconsistent*.

DL reasoning assumes an *Open World Assumption* [2], which does not imply a negation if the positive is not a deduction result, and vise versa. This means we cannot conclude that some policy (in format of an axiom) is unsatisfiable if it is not deduced from the KB. Therefore, given a concerned policy as an axiom A, the fourth AQ could be answered with a consistency checking as Formula (9), of the renewed KB as $\Sigma \models \{A\}$. The fifth AQ could be answered with a consistency checking of the renewed KB as $\Sigma \models \{\neg A\}$. The sixth AQ however, could not be answered with a single consistency check because of the open world assumption. It is answered as *irrelevant* only if the fourth and fifth queries are both answered *consistent*.

The last three AQ's are similar to AQ 4-6, with the only difference that instead of adding a new axiom A, an existing entity e, say a `SUBJECT`, `OBJECT` or `PERMISSION`, will be alternated by a 'new' entity with its attribute changed, i.e. all appearance of e will be replaced with an e'. Then the 'new' KB is checked in respect of Σ.

As mentioned for Formula (6), it could be decomposed into two steps.

1. Instance Retrieval: retrieve all the `OBJECT` o for the compound concept $\exists\ P^{-}.\{u\}$ with respect to Formula (8), which makes a set X;
2. Satisfiability Check: check the satisfiability of the axiom A in the form of $O \sqsubseteq X$ with respect to Formula (9).

Here in this subsection, nine patterns are discovered in Formulae (1 to 9). Different strategies will be applied on the patterns to enhance the run-time performance of the system.

4.2 Task Distribution

As is shown in Figure 2 in Section 3, the queries through interfaces are transformed by the Parse Engine PE and then transported to the other engines to process via interaction(s) with the KB. However, a general purpose reasoner such as [15], [7] and [13] does not provide good enough responses to *RelBAC* queries. To be precise, they cannot reason on *RelBAC* with the logic $\mathcal{ALCQIBO}$., not to say provide run-time answers. How to transform *RelBAC* into a form that could be operated by DL reasoners will not be covered for page limits. The aim of this paper is to make the system work at run-time.

The strategy is to distribute the queries to different engines, i.e. RE or QE. With the pattern clarification of *RelBAC* in Section 4.1, we distinguish the knowledge patterns into nine formulae (Formula 1 - 9). We see that only the CQ's relate to run-time queries. Moreover, they are all related to instances. Therefore, we design an algorithm to populate all the individuals inside the KB into possible concepts which named as ABoxing, and get to a theorem as the following.

Definition 1. *The ABox assertion set $\mathcal{A}$ of an ontology $\mathcal{O}$ is complete, if and only if any ABox assertion α implied by $\mathcal{O}$ is explicitly inside $\mathcal{A}$.*

$$\mathcal{A} = \{\alpha | \alpha \textit{ is an ABox assertion}, \mathcal{O} \models \alpha\}$$

Then we have the following theorem which gives that,

Theorem 1. *If the ABox assertion set of an ontology $\mathcal{A}$ is complete, then any answer by $\mathcal{O}$ to a CQ is the same as answered by $\mathcal{A}$.*

Proof. Given a CQ, the pattern of the query falls into one of the three form as Formula (5-6).

For a query in pattern of Formula (5), if $P(s, o)$ can be deduced from Σ then with Definition 1, it should be explicitly in $\mathcal{A}$, which will derive $P(s, o)$ apparently. If it is not deduced from Σ, either $\neg P(s, o)$ is deduced, or neither is deduced. For the first case, $\neg P(s, o)$ should be explicitly asserted in $\mathcal{A}$, therefore $P(s, o)$ cannot be derived; for the second case, without TBox axioms, $\mathcal{A}$ is only a set of assertions that could not deduce any fact that is not explicitly asserted in $\mathcal{A}$, then $P(s, o)$ cannot be derived.

For a query in pattern of Formula (6) if for each $o_i \in O$, $P(s, o_i)$ can be deduced from Σ, then s satisfies Formula (6); if $\mathcal{A}$ is complete then for each $o_i \in O$, $P(s, o_i)$ and $O(o_i)$ are explicitly asserted in $\mathcal{A}$, then can s is verified to fit Formula (6). Otherwise, if for all $o_i \in O$, there exists one $P(s, o_i)$ can not be deduced from Σ, it is then not asserted in $\mathcal{A}$, and cannot fit Formula (6).

An algorithm is proposed to populate individuals to concept in $\mathcal{A}$. In Algorithm 1, ABox assertion set and TBox axiom set are initialized on Line 1-2. The the loop in Line 3-10 computes all the necessary TBox axioms and explicitly adds them into $\mathcal{T}$. Then the second loop in Line 11-24 compute the ABox assertions according to the computed $\mathcal{T}$.

After population, $\mathcal{A}$ is complete. Each assertion in $\mathcal{A}$ could be coded into classical database record. Then the task to answer any CQ query is distributed to classical database queries. For the rest AQ, the task could be carried out with general purpose reasoners that support SWRL, which might be time-consuming, but acceptable for off-line administration.

Algorithm 1: ABox Population

Input: An ontology $\mathcal{O}$, consists of two parts namely the ABox assertions in a set $\mathcal{A}$ and the TBox axioms in a set $\mathcal{T}$.

Output: The complete ABox assertion set $\mathcal{A}$ of O

```
A ← O.A
T ← O.T
while T grows do
    for each a ∈T do
        if a is C ⊑ D then
            T +={C ⊑ D}
        if a is C ≡ D then
            T +={C ⊑ D, D ⊑ C}
        if a is C ⊓ D ⊑ E then
            T +={C ⊑ E, D ⊑ E}
while A grows do
    for each b ∈ A do
        if b is C(x) then
            for each a ∈ T do
                if a is C ⊑ D then
                    A +={D(x)}
        if b is R(x, y) then
            for each a ∈ T do
                if a is R ⊑ S or R ≡ S then
                    A +={S(x, y)}
                if a is ∀ R.⊤ ⊑ D then
                    A +={D(x)}
                if a is ∀ R⁻.⊤ ⊑ D then
                    A +={D(y)}
return A
```

5 Conclusion

RelBAC stands out of many other access control models for its rich expressiveness and formalism. Its logical complexity hinders its application in industry. We provide a framework to implement *RelBAC* in real-world software solutions. Queries that should be answered in run-time are distributed to classical database, that has complete ABox assertions coded into database records. Queries that could be answered off-line are distributed to DL reasoners with services of consistency checking, satisfiability checking, instance retrieval etc. With this combination in our framework, advantages of each query-answer mechanism could be taken and a practical implementation of *RelBAC* is foreseen.

References

1. Artale, A., Crispo, B., Giunchiglia, F., Turkmen, F., Zhang, R.: Reasoning about relation based access control. In: Xiang, Y., Samarati, P., Hu, J., Zhou, W., Sadeghi, A.R. (eds.) NSS. pp. 231–238. IEEE Computer Society (2010)
2. 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, New York, NY, USA (2003)
3. Damiani, M.L., Bertino, E., Catania, B., Perlasca, P.: Geo-rbac: A spatially aware rbac. ACM Transactions on Information and System Security 10(1) (2007)
4. Ferraiolo, D.F., Cugini, J.A., Kuhn, D.R.: Role-Based Access Control (RBAC): Features and Motivations. In: 11th Annual Computer Security Applications Proceedings (1995)
5. Giunchiglia, F., Zaihrayeu, I.: Lightweight ontologies. In: Encyclopedia of Database Systems. Springer (2008)
6. Giunchiglia, F., Zhang, R., Crispo, B.: Relbac: Relation based access control. In: SKG '08: Proceedings of the 2008 Fourth International Conference on Semantics, Knowledge and Grid. pp. 3–11. IEEE Computer Society, Washington, DC, USA (2008)
7. Haarslev, V., Möller, R.: Racer: A core inference engine for the semantic web. In: Proceedings of the 2nd International Workshop on Evaluation of Ontology-based Tools (EON2003), located at the 2nd International Semantic Web Conference ISWC 2003, Sanibel Island, Florida, USA, October 20. pp. 27–36 (2003)
8. Kuhn, D.R., Coyne, E.J., Weil, T.R.: Adding attributes to role-based access control. IEEE Computer 43(6), 79–81 (2010), http://dblp.uni-trier.de/db/journals/computer/computer43.html#KuhnCW10
9. Lutz, C., Sattler, U.: The complexity of reasoning with boolean modal logics. In: Wolter, F., Wansing, H., de Rijke, M., Zakharyaschev, M. (eds.) Advances in Modal Logics Volume 3. CSLI Publications, Stanford (2001)
10. Lutz, C., Walther, D.: Pdl with negation of atomic programs. Journal of Applied Non-Classical Logic 15(2), 189–214 (2005)
11. Oh, S., Sandhu, R.S.: A model for role administration using organization structure. In: SACMAT. pp. 155–162 (2002)
12. Schmidt, R.A., Tishkovsky, D.: Using tableau to decide expressive description logics with role negation. In: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC + ASWC. Lecture Notes in Computer Science, vol. 4825, pp. 438–451 (2007)
13. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical owl-dl reasoner. Submitted for publication to Journal of Web Semantics. (2003)
14. SWRL: Http://www.w3.org/Submission/SWRL/
15. Tsarkov, D., Horrocks, I.: Fact++ description logic reasoner: System description. In: Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2006). Lecture Notes in Artificial Intelligence, vol. 4130, pp. 292–297. Springer (2006)
16. Zhang, R., Artale, A., Giunchiglia, F., Crispo, B.: Using description logics in relation based access control. In: Proceedings of the DL Home 22nd International Workshop on Description Logics (DL 2009), Oxford, UK, July 27-30, 2009 (2009)

Novel Integrated Framework for Crowd Simulation

Qingge Ji* and Yugang Han

School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China
issjqg@mail.sysu.edu.cn

Abstract. In this paper, two intuitive and highly efficient solutions are proposed for global planning and local avoidance. We introduce guide and repel vectors to study global planning, which generates a steady and smooth navigation field through a simple and efficient bilinear interpolation method. In addition, this paper proposes a novel velocity-based approach to simulate the local avoidance of agents based on least-effort principle. During the local avoidance phase, humans slightly adjust their motions, so that the energy required to perform a step becomes minimal. The two solutions are integrated into one system, which finally simulates the natural-looking navigation and interaction behavior of agents.

Keywords: Crowd Simulation, Global Planning, Navigation Fields, Local Avoidance, Least effort

1 Introduction

As virtual reality technology develops, crowd simulation technology is paid increasing attention. According to different modeling granularities, existing crowd models can be generally classified into two categories, namely, macroscopic and microscopic approaches. The former models a crowd as continuous flow of fluid [1]. This technology is mainly useful in large and dense crowds but basically neglects the features of individuals. The latter models a crowd as a collective of homogeneous/heterogeneous entities with interactions among them, and the representative approaches include entity-based and agent-based. Individuals are modeled as a set of homogenous entities in the entity-based approach. A typical example of this approach is Helbing's social force model (SFM) [2]. The agent-based approach models each individual in a crowd as an intelligent and autonomous agent [3], in which each agent perceives its own state and reacts to dynamic entities in its neighborhood. The microscopic approach models are flexible, such that adding physical, social, and psychological factors can simulate various interactive behavior. As a result, these models are the most popular ones. However, their computing cost is high. Jiang et al.[4] presented a semantic model for representing the complex environment, where the semantic information is described with a geometric level, a semantic level and an application level. The model promotes the interactions between pedestrians and the environment.

Kraayenbrink et al.[5] proposed semantic crowds that allowed one to re-use the same population for virtual environment.

Main Contribution: Based on previous research, two intuitive and highly efficient solutions are proposed in this paper for global planning and collision avoidance.

We introduce guide and repel vectors to study global planning, which generates a steady and smooth navigation field through a simple and efficient bilinear interpolation method. In addition, we propose a novel velocity-based approach to simulating the collision avoidance of agents through the observation of human behavior in avoiding dynamic obstacles in real life.

2 Related Work

In this section, we briefly discuss prior literature on global planning and local avoidance, which are the two key issues in crowd modeling technology.

Global Planning: To navigate a complex environment, a high-level path planning technology is needed. The most popular crowd navigation technologies include graph search and potential fields. Graph-based algorithms are widely used in global planning [6]. Pettre et al. [7] proposed a graph structure that decomposes a space into multilayered terrains to support fast graph search for multiple characters. Bandi et al. [8] extended A* algorithm to a 3D space and reproduced many interesting navigation behaviors. Roadmaps [9] and Voronoi diagrams [10] are recently introduced to crowd navigation. Potential fields are extensively studied in robot motion planning [11]. Dapper et al. [12] introduced harmonic function to generate potential fields; thus, they would not fall into the local minimum and could simulate various navigation behaviors by adjusting the parameters in the function. Moreover, many researchers have directly attempted to govern navigation by computing velocity fields based on environment description [13], designing velocity fields manually [14], or capturing the velocity fields from videos and user inputs [15]. Our global planning algorithm is inspired by [13]. We introduce two types of vectors, namely, repel and guide vectors. An efficient bilinear interpolation method is used to obtain smooth navigation fields.

Local Avoidance: Collision should be avoided locally by adjusting movements when other agents become sufficiently close. Many local avoidance approaches have been proposed, including particle force interaction [16], geometric [17], and synthetic-vision models [18]. Many researchers have introduced velocity-based methods for collision avoidance recently. Paris and Pettre et al. [19] proposed a predictive approach and resolved potential collisions successfully. Karamouzas and Overmars [20] proposed a velocity-based approach by analyzing experimental data and extended this approach to small groups [21]. Koh and Zhou [22] introduced a collision avoidance framework called relative frame. According to the duality property of the relative frame and other constraints, they selected a collision-free velocity for an agent. Our local avoidance algorithm is inspired by the work of Koh and Zhou. We use a modified relative frame to predict the potential collision and select an optimal velocity for an agent. However, unlike Koh and Zhou, we adopt the least-effort principle and eventually obtain a realistic and natural-looking result.

3 Global Path Planning

3.1 Environment Decomposition and Organization

To compute a global path to the goal for each agent, we decompose the environment into grids, which have different size and are represented by rectangles. When static obstacles are dense, our method will subdivide the environment until each mixed grid is almost occupied by obstacles; when static obstacles are sparse, our decomposition method roughly divides the environment into several grids, then merges the empty grids, and forms a large empty area.

We use a four-connected graph to organize the empty grids. The connective graph is defined to be the graph that has a vertex for each grid and an edge between two vertices only if the corresponding grids share a segment on their boundaries. A path over this graph is computed, such that following the path from any vertex leads to the vertex corresponding to the grid containing the goal state. The resulting directed graph defines a successor for every grid, except the goal grid. The successor of a grid is the next grid on the path to the goal grid. Each grid with a successor is termed as an intermediate grid, and the intermediate grid has only one successor. Specially, the goal grid has no successor. See Figure 1 for an illustration.

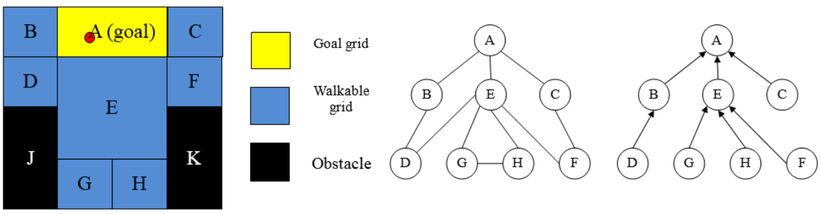


Fig. 1. Environment decomposed into grids and the corresponding connectivity and directed graphs.

3.2 Repel and Guide Vectors

We assume that each grid has four adjacent grids (because the graph is four-connected). A grid must be set as the successor of the grid. The shared boundary is called exit face, and the others are called repel faces. See Figure 2 for an illustration. To obtain a proper transition from the current grid to successor, we introduce two types of vector fields, i.e., those corresponding to grids in the decomposition, which we call guide vector fields, and those corresponding to faces, which we call repel vector fields. A guide vector field guides an agent through the grid to the exit face, which leads to the successor grid. Repel vector fields prevent an improper grid transition, i.e., a transition from the current grid to a grid that is not the successor is prohibited. For the repel vector on repel faces, its direction is orthogonal to the face and points inward. For the repel vector on the exit face, its direction is orthogonal to the exit face and points outward. The guide vector fields always point toward the exit face. In the case of the goal grid, all repel and guide vector fields point inward to the goal state.

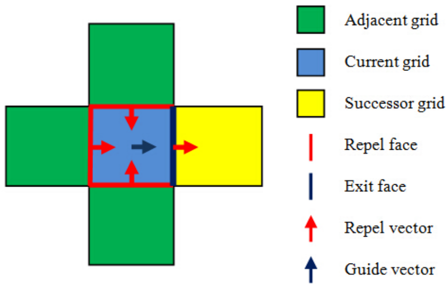


Fig. 2. Illustration for repel face, exit face, repel vector, and guide vector.

The different sizes between adjacent grids pose a difficulty in choosing the appropriate repel or guide vector fields. Zhang et al. [23] proposed a method to resolve this problem.

3.3 Vector Interpolation

To obtain a smooth transition from the current grid to successor for an agent, an efficient and simple bilinear interpolation method is used to compute the final repel vector $\boldsymbol{V_{repel}}$ (Figure 3).We assume that the position of agent(x_i,y_i) is in grid C=$\{(x_1, y_1), (x_2, y_2)\}$, and its successor is S=$\{(x_3, y_3), (x_4, y_4)\}$, where $\{(x_1, y_1), (x_2, y_2)\}$ and $\{(x_3, y_3), (x_4, y_4)\}$ represent the upper left and lower right vertex coordinates of C and S, respectively. The repel vector set of grid C is $\boldsymbol{F}$=$\{\boldsymbol{f_0}, \boldsymbol{f_1}, \boldsymbol{f_2}, \boldsymbol{f_3}\}$.

$$\boldsymbol{V_{repel}} = \frac{x_2 - x_i}{x_2 - x_1} * \boldsymbol{f_0} + \frac{x_i - x_1}{x_2 - x_1} * \boldsymbol{f_2} + \frac{y_2 - y_i}{y_2 - y_1} * \boldsymbol{f_1} + \frac{y_i - y_1}{y_2 - y_1} * \boldsymbol{f_3} \quad (1)$$

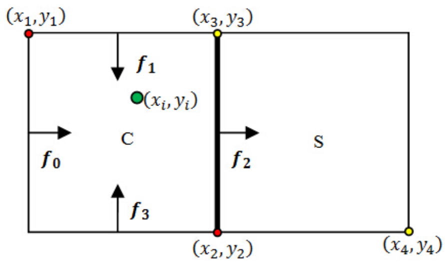


Fig. 3. Computation of the final repel vector $\boldsymbol{V_{repel}}$.

Considering that the grid size might differ, two cases are considered for $\boldsymbol{f_2}$. Figure 4 shows that when the current position of agent (x_i, y_i) locates below the green-dotted line, $\boldsymbol{f_2}$=$\boldsymbol{f_{virtual}}$, when (x_i, y_i) locates above the green-dotted line, $\boldsymbol{f_2}$=$\boldsymbol{f_{21}}$. $\boldsymbol{f_{virtual}}$ represents the repel vector on the virtual face, and $\boldsymbol{f_{21}}$ represents the repel vector on the exit face.

$$\boldsymbol{f_2} = \begin{cases} \boldsymbol{f_{21}} & y_i > y_3 \wedge y_i < y_4 \\ \boldsymbol{f_{virtual}} & y_i \geq y_4 \wedge y_i \leq y_2 \end{cases} \tag{2}$$

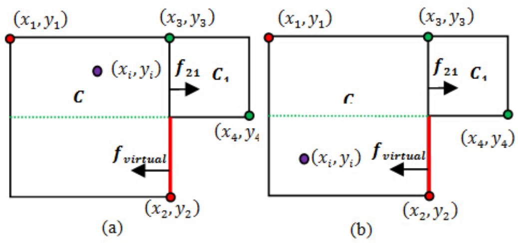


Fig. 4. Selection of $\boldsymbol{f_2}$ in two cases.

Assuming that the guide vector is $\boldsymbol{V_{guide}}$, we calculate the linear interpolation between $\boldsymbol{V_{repel}}$ and $\boldsymbol{V_{guide}}$, and obtain the navigation vector at (x_i, y_i), denoted as $\boldsymbol{V_{nav}}$.

$$\boldsymbol{V_{nav}} = \alpha * \boldsymbol{V_{repel}} + \beta * \boldsymbol{V_{guide}} \tag{3}$$

We suppose that $\alpha = 0.5$ and $\beta = 0.5$. We can calculate the navigation vector of each spot in the configuration space using Equation (3). Disregarding other agents, each agent can move step by step along the direction of $\boldsymbol{V_{nav}}$ to the goal state. Figure 5(a) shows an example for the navigation fields, and Figure 5(b) shows the path of an agent moving from the initial point to the goal state. No steep turn exists in the corners, and the whole path is smooth, which vividly simulates the human behavior when turning in our real life.

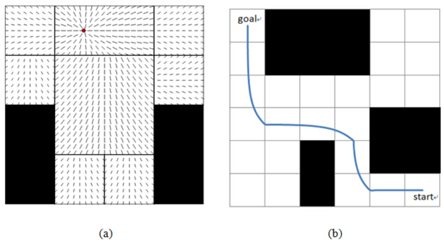


Fig. 5. (a) Example for the navigation fields. The black rectangle denotes obstacle, and the red point denotes the goal state. (b) Path for an agent from the initial position to the goal state.

4 Local Collision Avoidance

Two main challenges occur in local collision avoidance, namely, collision prediction and collision avoidance. In this section, we describe our collision avoidance model.

4.1 Problem Formulation

In our problem setting, we are given a virtual environment where n agents PN=$\{P_1, \ldots, P_n\}$ have to navigate toward their specified goal without colliding with the environment and with one another. For simplicity, we assume that each agent moves on a plane and is modeled as a disc with radius r_i, and its personal space is modeled as a disc with radius ρ_i. At a fixed time t, the agent P_i is at the position $\boldsymbol{x_i(t)}$, defined by the disc center, and moves with velocity $\boldsymbol{v_i(t)}$. This velocity is limited by a maximum speed u_i^{max}, i.e., $\|\boldsymbol{v_i(t)}\| \leq u_i^{max}$. For notational convenience, we will not explicitly indicate the time dependence.

In every simulation step, the agent P_i has a desired velocity $\boldsymbol{v_i^{des}(t)}$, whose orientation is $\boldsymbol{V_{nav}}$, which have been computed in Section 3, and magnitude is u_i^{des}, which is closely related to the crowd density ρ according to Fang et al. [24].

$$\boldsymbol{v_i^{des}} = u_i^{des} * \frac{\boldsymbol{V_{nav}}}{\|\boldsymbol{V_{nav}}\|} \tag{4}$$

$$u_i^{des} = \begin{cases} u_i^{max} & \rho \leq \rho_{min} \\ u_i^{min} + \frac{\rho - \rho_{min}}{\rho_{max} - \rho_{min}} * (u_i^{max} - u_i^{min}) & \rho_{min} < \rho < \rho_{max} \\ \bar{u} & \rho \geq \rho_{max} \end{cases} \tag{5}$$

In the above equations, ρ_{min} and ρ_{max} are the minimum and maximum crowd density thresholds, respectively. $\bar{u}$ is the average speed of all agents, which are in the vision range of P_i 's vision range.

4.2 Collision Prediction

An agent configuration is defined by its position and velocity. Koh and Zhou proposed a relative frame model for collision prediction. Source agent is denoted as the agent that avoids a target agent. Figure 6 shows the relative frame between a source agent and a target agent, where v_r is the relative velocity between the source and target agents; θ_s and θ_g are the orientation of the source and target agents, respectively; θ_r is the relative orientation between the source and target agents. $R_g = r_g + \rho_s$, it means that the target agent should not invade the personal space of the source agent.

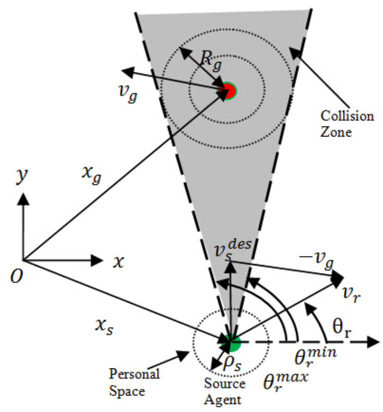


Fig. 6. Relative frame.

The collision zone is defined as a region of space where the source agent should prevent collision with the target agent, i.e., collision is predicted in future if

$$\theta_r^{min} \leq \theta_r \leq \theta_r^{max} \tag{6}$$

and if the two agents do not change their speed and orientation.

When a collision has been predicted, we then compute the time to collision (ttc); if ttc is less than a certain anticipation time t, the target agent is inserted into a set of agents that are on the collision course with the source agent. In real-life, an individual tries to avoid a limited number of other pedestrians, usually those that are on the collision course with him in the coming short time. Similarly, the source agent tries to evade N agents with which will collide first. In our implementation, N is less than 4.

4.3 Collision Avoidance

The least-effort principle originates from the field of psychology and states that given different possibilities of actions, people select the one that requires the least effort [25]. Based on least-effort theory, many systems for crowd simulation have been proposed [26], [27]. However, all these approaches aim to control the macroscopic (global) behavior of virtual humans, whereas our focus is on the local interactions of individuals. Based on the least-effort principle, we therefore hypothesize that an individual, upon interacting with other individual, tries to resolve potential collisions immediately by slightly adapting his motion. The individual will adjust his trajectory in advance, trying to reduce the interactions with the other walker. We describe our local avoidance algorithm below.

We first retrieve a set of candidate relative orientation O_r, such that the orientation of relative velocity can be selected to resolve the collision with the agents who are on the collision course. According to condition (6), the collision can be avoided if the source agent selects a new relative velocity $\boldsymbol{v_r^{new}}$, that satisfies the condition

$$\neg(\theta_r^{min} \leq \theta_r^{new} \leq \theta_r^{max}) \tag{7}$$

To avoid unrealistic orientation deviate, we bound the max angle deviation θ_i^{max} to $\frac{\pi}{2}$. We can compute O_r by combining condition (7) and θ_i^{max}.

We then retrieve the set of candidate relative speed U_r. When O_r is determined, the max relative speed $u_r^{max}{=}\left\|v_i^{des}\right\|{+}\|v_j\|$ and the min relative speed $u_r^{min}{=}\left|\left\|v_i^{des}\right\| - \|v_j\|\right|$.

Having retrieving O_r and U_r, we select an optimal pair $\boldsymbol{P}{=}(u_r{,}\theta_r)$, where $u_r \in U_r \wedge \theta_r \in O_r$, so that the expenditure energy for the source agent is minimum. See Figure 7(a) for an illustration.

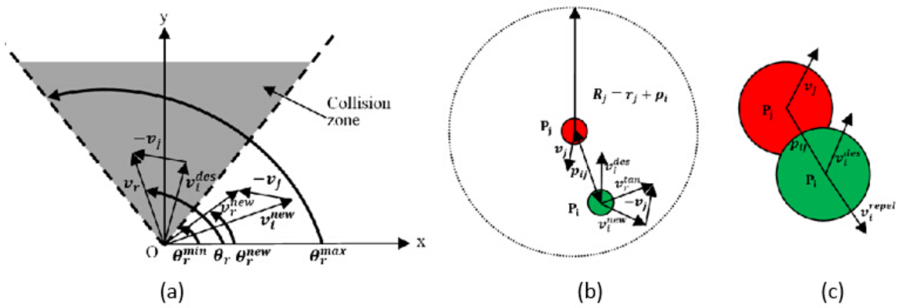


Fig. 7. (a)Selection of an optimal relative velocity for the source agent.(b)(c)Two cases for imminent collision:case(b)One agent enters into the personal space of other but they are not overlapping yet.case(c)Two agents have overlapped

$$\boldsymbol{v_i^{new}} = \boldsymbol{v_r^{new}} + \boldsymbol{v_j} \tag{8}$$

$$\Delta u_i = \left|\|\boldsymbol{v_i^{new}}\| - \left\|\boldsymbol{v_i^{des}}\right\|\right| \tag{9}$$

$$\Delta\theta_i = \arccos\frac{\boldsymbol{v_i^{new}} \cdot \boldsymbol{v_i^{des}}}{\|\boldsymbol{v_i^{new}}\| * \left\|\boldsymbol{v_i^{des}}\right\|} \tag{10}$$

In the above equations, Δu_i is the value for speed changed, and Δ_i is the angle deviation of the source agent. The cost function is

$$f(u_r, \theta_r) = \alpha * \frac{\Delta u_i}{u^{max}} + \beta * \frac{\Delta \theta_i}{\theta^{max}} \tag{11}$$

where u^{max}=1.5m/s is the maximal value for speed changed, and $\theta^{max}=\frac{\pi}{2}$ is the maximum angle deviation. The constants α and β define the weights of specific cost terms and can vary among the agents to simulate a wide variety of avoidance behavior.

Computing the minimum value of Equation (11) is time-consuming. Thus, we restrict the domain O_r to a discrete set of orientation samples (the default size of the discretization step is set to 0.01π). Similarly, we discretize the domain U_r into a set of adjacent speed samples (the default distance between adjacent samples is set to 0.05). Assuming that the discretized set of O_r is $\mathbf{O}_r$ and that of U_r is $\mathbf{U}_r$, then the set of admissible relative velocity is

$$FAV_r = \{u_r \theta_r \mid u_r \in \mathbf{U}_r \wedge \theta_r \in \mathbf{O}_r\} \tag{12}$$

The discretized cost function is

$$v_r^{new} = \underset{\boldsymbol{V}^{cand} \in FAV_r}{\operatorname{argmin}} \left\{ \alpha * \frac{\left| \left\| v^{cand} + v_j \right\| - \left\| v_i^{des} \right\| \right|}{u^{max}} + \beta * \arccos \frac{(v^{cand} + v_j) \cdot v_i^{des}}{\left\| v^{cand} + v_j \right\| * \left\| v_i^{des} \right\|} \right\} \tag{13}$$

Having retrieving v_r^{new}, the optimal new velocity for the source agent is easy to compute. We then update the source agent position into

$$\boldsymbol{x_i^{new}} = \boldsymbol{x_i} + \boldsymbol{v_i^{new}} * \Delta t \tag{14}$$

4.4 Resolve Imminent Collision

We divide imminent collision into two cases (Figure 7(b)(c)). In case (b), we introduce the concept of relative tangential velocity, which is equivalent to applying a tangential force to separate the two agents. In case (c), we introduce the concept of repel velocity, which is equivalent to applying a repulsive force to separate the two agents immediately.

4.5 Avoiding Static Obstacles

An agent A_i also needs to avoid colliding with the static obstacles of the environment. In our simulations, such obstacles are modeled as axis aligned boxes. Collisions are resolved by following an approach similar to the one described above.

We first retrieve the nearest obstacles of the agent A_i that are inside the visual field of the agent. We then compute the maximum and minimum orientations among the vectors lined from the current position of A_i 's to each vertex of the convex polygon obstacle. The maximum and minimum orientations are θ_r^{max} and θ_r^{min}, respectively, which have been discussed above. Finally, we use a least-effort criterion to select an optimal velocity for the agent A_i.

5 Experimental Results

We test our approach against a wide range of scenarios. These scenarios range from the simple interactions between pairs of agents to more challenging and large test cases as follows:

- Squeeze: Two agents have to avoid a head-on collision while walking in an opposite direction (Figure 8(a)).
- Overtake: An agent moves down a hallway and encounters a slower agent in front (Figure 8(b)).
- Square: Four agents are placed on the vertex of a square and have to walk toward their diagonal position (Figure 8(c)).
- Complex environment: Three hundred agents walk through an environment filled with many obstacles and have to evacuate from the exit (Figure 8(d)).

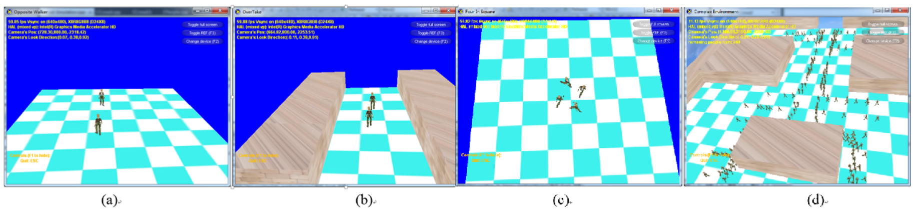


Fig. 8. Scenarios: (a)-(c) interactions in simple environments; (d) three hundred agents evacuate from an obstacle-filled environment.

6 Conclusion

In this paper, we present a novel integrated framework for navigation and interaction behavior. A creative global path planning algorithm and a bilinear interpolation method were used to compute the navigation fields. A least-effort criterion was also employed in the local avoidance to achieve realistic local movements.

7 Acknowledgement

This research was supported by National Natural Science Foundation of China (No. 60473109), NSFC-Guangdong Union Foundation of China (No.U0735001). We would like to thank the anonymous reviewers for their suggestions and comments.

References

1. Treuille A, Cooper S, Popovi? Z. Continuum crowds[C]. ACM Transactions on Graphics (TOG). ACM, 2006, 25(3): 1160-1168.
2. Helbing D, Molnar P. Social force model for pedestrian dynamics[J]. Physical review E, 1995, 51(5): 4282-4286.)
3. Shao W, Terzopoulos D. Autonomous pedestrians[C]. Proceedings of the 2005 ACM SIGGRAPH/Eurographics, 2005: 19-28.
4. Hao Jiang, et al., A Semantic Environment Model for Crowd Simulation in Multilayered Complex Environment. VRST 2009, Kyoto, Japan, November 18 C 20, 2009. pp191-198
5. Kraayenbrink N, Kessing J, Tutenel T, et al. Semantic crowds. Entertainment Computing[J], 2014(5):297-312
6. Paris S, Donikian S, Bonvalet N. Environmental abstraction and path planning techniques for realistic crowd simulation[J]. Computer Animation and Virtual Worlds, 2006, 17(3-4): 325-335.
7. Pettre J, Laumond J P, Thalmann D. A navigation graph for real-time crowd animation on multilayered and uneven terrain[C]. First International Workshop on Crowd Simulation. 2005, 43(44): 194-202.
8. Bandi S, Thalmann D. Space discretization for efficient human navigation[C]. Computer Graphics Forum. Blackwell Publishers Ltd, 1998, 17(3): 195-206.
9. Bayazit O B, Lien J M, Amato N M. Better Group Behaviors in Complex Environments using Global Roadmaps[J]. Artificial Life 8, 2003: 362-371.
10. Sud A, Andersen E, Curtis S, et al. Real-time path planning in dynamic virtual environments using multiagent navigation graphs[J]. Visualization and Computer Graphics, IEEE Transactions on, 2008, 14(3): 526-538.
11. Philippsen R, Siegwart R. An interpolated dynamic navigation function[C]. Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. IEEE, 2005: 3782-3789.
12. Dapper F, Prestes E, Idiart M A P, et al. Simulating pedestrian behavior with potential fields[M]. Advances in Computer Graphics. Springer Berlin Heidelberg, 2006: 324-335.
13. Lindemann S R, LaValle S M. Simple and efficient algorithms for computing smooth, collision-free feedback laws over given cell decompositions[J]. The International Journal of Robotics Research, 2009, 28(5): 600-621.
14. Park M J. Guiding flows for controlling crowds[J]. The Visual Computer, 2010, 26(11): 1383-1391.
15. Patil S, Van Den Berg J, Curtis S, et al. Directing crowd simulations using navigation fields[J]. Visualization and Computer Graphics, IEEE Transactions on, 2011, 17(2): 244-254.
16. Karamouzas I, Heil P, van Beek P, et al. A predictive collision avoidance model for pedestrian simulation[M]. Motion in Games. Springer Berlin Heidelberg, 2009: 41-52.
17. Van den Berg J, Lin M, Manocha D. Reciprocal velocity obstacles for real-time multi-agent navigation[C]. Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on. IEEE, 2008: 1928-1935.
18. Ondřej J, Pettr J, Olivier A H, et al. A synthetic-vision based steering approach for crowd simulation[J]. ACM Transactions on Graphics (TOG), 2010, 29(4): 123-131.
19. Paris S, Pettr J, Donikian S. Pedestrian reactive navigation for crowd simulation: a predictive approach[C]. Computer Graphics Forum. Blackwell Publishing Ltd, 2007, 26(3): 665-674.

20. Karamouzas I, Overmars M. Simulating human collision avoidance using a velocity-based approach[C]. Workshop in Virtual Reality Interactions and Physical Simulation. The Eurographics Association, 2010: 125-134.
21. Karamouzas I, Overmars M. Simulating and evaluating the local behavior of small pedestrian groups[J]. Visualization and Computer Graphics, IEEE Transactions on, 2012, 18(3): 394-406.
22. Wee Lit Koh and SuiPing Zhou. Modeling and Simulation of Pedestrian Behavior in Crowded Places. ACM Transactions on Modeling and Computer Simulation, 2011, 21(3): 20-42.
23. Zhang L, LaValle S M, Manocha D. Global vector field computation for feedback motion planning[C]. Robotics and Automation, 2009. ICRA'09. IEEE International Conference on. IEEE, 2009: 477-482.
24. Fang Z, Lo S M, Lu J A. On the relationship between crowd density and movement velocity[J]. Fire Safety Journal, 2003, 38(3): 271-283.
25. George K. Zipf. Human behavior and the principle of least effort[J]. Addison-Wesley, 1949. 3.
26. Safonova A, Hodgins J K. Construction and optimal search of interpolated motion graphs[C]. ACM Transactions on Graphics (TOG). ACM, 2007, 26(3): 106-116.
27. Guy S J, Curtis S, Lin M C, et al. Least-effort trajectories lead to emergent crowd behaviors[J]. Physical Review E, 2012, 85(1): 016110.

Position on Interoperability Everywhere under IoT-ARM

Adriano Tavares[1], Fausto Giunchiglia[2], Hao Xu[3], and Yanchun Liang[3]

[1]DEI, University of Minho, 4800058, Portugal
[2]DISI, University of Trento, 38100, Italy
[3]CCST, Jilin University, Changchun, 130012, China
atavares@dei.uminho.pt
fausto@disi.unitn.it
{xuhao,ycliang}@jlu.edu.cn

Abstract. This position paper argues that accelerating the use of IoT-ARM (Architectural Reference Model) on new IoT-Systems' realizations requires semantic interoperability to more than architectural, device and connectivity levels but also at tool-, system stack-, language- and workflow management-level. In doing so, an IoT-ARM ontology is proposed which extends the conceptual model of IoT-ARM method with highly cohesive Methodology Mapping, Big Data Analytic, and Architecture Implementation Roadmap facets while leveraging cross- and intra-language interoperability.

Keywords: Semantic Interoperability, Multi-Lingual Interoperability, Big Data, Open Data, IoT, Ontology, IoT-ARM.

1 Introduction

Currently, we are entering the Internet of Things (IoT)-age with IoT-systems consisting of components like sensing, heterogeneous access, information processing and, applications and services. IoT movement relies on pervasive connectivity and intelligently connects humans, devices, and systems by integrating multiple technologies under a unified management platform. Architecturally, IoT follows a serviced-oriented model and it can be split into four tiers. Thing-tier for sensing and transmission, Intelligent System-tier (i.e., Fog-tier) for early-life data analysis, aggregation and transmission, Cloud-tier for early-life or at-rest science-data analysis and storing and, Application-tier for user access and control.

Driven by the heterogeneity of IoT-related ecosystems, several problem spaces have been identified like connectivity, architecting process, big data analytics, device intelligence and data technologies that must be overcome to achieve mainstream IoT adoption. As IoT spans various industries and use cases, embedded processing will demand scalable strategies while a limited scope of standards will coexist for a long time to come as one size will not fit all [1, 2]. IoT-ARM emerged as a possible answer to the IoT multiplicity issue and it started by creating the IoT Reference Model (IoT-RM) to promote a common understanding, followed by the IoT Reference Architecture (IoT-RA) that describes essential building blocks and design choices to deal with conflicting quality attributes like functionality, performance, deployment and security [3, 4]. IoT-ARM approaches a loosely-coupled interoperability at connectivity- and semantic-level and it relies on the semantic technology to apply interoperability at architectural-level through the IoT Domain Model and IoT Information Model. It also addresses connectivity interoperability using a service-oriented communication model leveraged on the ISO OSI 7-layer model and it aims at highlighting those peculiar interoperability aspects inherent to the interoperation among different stacks, which are called interoperability features. Furthermore, it builds variation points into the software, and uses standard extension points, e.g., using standardized protocols and gateways to enable brownfield deployment [3].

From our point of view, IoT-ARM presents some drawbacks that are decelerating its use on new IoT-Systems' realizations such as:

- It only partially addresses device technology issues as its focus is only on the software stack (i.e., it does not address the whole system stack).
- Its Architecting Process is too generic and so, several projects instead of following IoT-ARM from the ground up try to show at the end of their architecture realizations how do they map to the IoT-ARM (e.g., [5]).
- It does not explicitly promote the interplay of IoT with Big Data as its main focus goes to IoT applications for track, command, control and route (TCC&R) purposes.

2 Our Approach

Our approach aims at designing a modular ontology to assist in IoT-ARM Architecting Process (i.e., IoT-ARM ontology) as an instance of UKC [6, 7]. The UKC domain, concept, and entity type cores will be extended by terminology specific to a new field of study, in order to assist semantically the IoT-ARM Architecting Process which is denominated IoT-ARM ontology. Basically, IoT-ARM ontology is a synthesis of ontologies proposed in [8, 9], extended with the Methodology Mapping facet to accommodate IoT-ARM methodology agnosticism and Big Data Analytic facet to enable interplay of IoT with Big Data. Extending IoT-ARM with the idea of Everywhere Interoperability based on a scalable semantic schema built as an instance of UKC will leverage:

- ***Multi-Lingual Interoperability*** at both cross-language and intra-language levels. Cross-language interoperability among several languages will enable their entrance into the "Open Big Data Age" while intra-language interoperability allows the use of multiple terms denoting the same concepts or more specific/general terms.
- ***Declarative design of a semantic and scalable whole design stack*** for easy customization at each IoT tier and better addressing the increasing

intelligence, security, safety, communication, timeliness, area, and power issues. The Architectural Description facet is extended with specific sub-facets representing the required knowledge for co-design strategies and propagation effects among the stack layers while the Architecture Design facet with a technological design flow sub-facet for the whole stack design. All entities including tools (e.g., design, simulation, synthesis tools) are declaratively and semantically tagged for the purpose.

- ***Semantic collaborative system design chain*** according to known workflow reference model dictated by industry horizontalization. The Architecture Implementation Roadmap facet is extended with a business collaboration workflow sub-facet embedding industry chain management knowledge.
- ***Big Data Analytic Reference Architecture*** to model Big Data Analytics space problem in terms of several levels of heterogeneity involved on its architecting and design process, at analytical types, use cases scenarios, location of analytic technology, analytic techniques, type of actionable intelligence and visualization, sources and type of data, technological platforms, spectrum of analytical workloads, etc.

To support the proposed multi-level interoperability, we are using SCROLL NLP and UKC frameworks developed by the KnowDive team at University of Trento. Following the UKC's so-called faceted approach to ontological modeling, the IoT-ARM ontology is extended by a large number of concepts (e.g., Sw, Hw and Simulation components, views, tactics, design choice, perspectives, quality attributes, system, design stack, design flow), entities (e.g., Linux, Windows, FreeRTOS, OSGi framework, ARM Cortex-M3, MPSoC, Hadoop, Oracle, OpenStack, VMWare, Cassandra, Simulink, Modelica), and highly cohesive facets (e.g., Methodology Mapping, Architectural Description, Architectural Requirements, Architecture Design, Architecture Implementation Roadmap). SCROLL NLP has been extended to support several languages by collecting linguistic resources, adapting and integrating them into a processing pipeline.

Following standardization on ontology leverages abilities of a gradually growing IoT-ARM environments (i.e., by skipping out of the "Standard War"), mapping of different vendors/providers technologies to the IoT-ARM, tooling enablement from different vendors/providers to the IoT-ARM environment and ready-made ecosystem of partners, thus enabling IoT-Systems' realizations of several and different use cases scenarios. After populating IoT-ARM ontology with several catalogs of entities divided by categories and semantically tagged with their properties and constraints, tools for managing templates of system stack and development flow through functionalities such as creation, instantiation, configuration, validation and deployment are designed and implemented. A team of experts in a given IoT application domain can create templates for specific applications and use cases scenarios (i.e., enabling some kind of application guidance) and populate the associated catalogs after validation. Furthermore, they specify mapping strategies described as semantics rules to associate tactics to design choices and then add them to the IoT-ARM tactic and design choice catalog. Later a user can instantiate existing template seeds from the catalog for his/her new IoT system realization. If a template contains abstract components/tools, then the component/tool catalog is queried to find a valid bind.

Several reasoners will be implemented to: (1) reason about the design space, (2) assist in the creation of design flows and system stacks, (3) reason about components' constraints and tool's characteristics and propagate them through the development flow and system stack structures and (4) reason about the matching between a development flow and a system stack. After reasoning about the design space and identifying all valid instances of system stack, a virtual prototyping environment can be built to explore such solution space. Such environment is a specialization or specific instance of the development flow to carry out mixed-simulation, including the dynamics of the physical process using tools such as Simulink or Modelica which are possible entities of IoT-ARM ontology populated into the catalog of tools. According to the IoT-ARM architectural description, a reasoner will be provided for each qualitative requirement to find a perspective associated with it and also to select tactics based on the functional requirements and architectural constraints. Finally, semantics rules are proposed to prioritize the way that perspectives will be applied to views as not all perspectives have equal effect on all views.

3 Opportunities Addressed

By extending IoT-ARM ontology to be more focused on main IoT current issues faced by new IoT-related system realizations, such as, technical, architecture, hardware, privacy and security, standard and business challenges, the conceptual model of IoT-ARM method is improved by tackling its poor methodological completeness and application guidance as pointed in [10] while promoting:

- Multi-sourced and multi-lingual big data analytics supporting real-time open data.
- Interoperability at stakeholder-level while mitigating IoT market fragmentation and industry disjointed tooling ecosystem.
- 3C (Computing, Control and Communication)- and 3S (Scalability, Security and Safety)-convergences in the new 'Cloud + Edge' computing paradigm through a holistic collaborative design chain approach encompassing end-device, connectivity layer, gateway, and services running in the cloud with design metrics or quality attributes factored in at every level.
- Some level of automation and application guidance to the IoT-ARM architecting process by approaching semantic whole system stack as well as semantic design flow with interoperability at tool-level.

Acknowledgments. The authors would like to acknowledge the assistance of Drs. Gabor Bella, Feroz Farazi and Enzo Maltese at the University of Trento, and PhD students Sandro Pinto and Paulo Garcia at the University of Minho, for their useful discussions and efforts on writing related proposals. The first author is grateful to the support of FCT - Fundação para a Ciência e Tecnologia in the scope of the project: PEst-UID/CEC/00319/2013. The last two authors are grateful to the support of the NSFC (61272207, 61300147, 61472158).

References

1. Karimi, K., Atkinson, G.: What the Internet of Things (IoT) Needs to Become a Reality. In: Freescale White Paper (2013)
2. Xing, Z.H., Zhong, Y.F.: Internet of Things and its future. In: Huawei Communicate, Issue 54 (2010)
3. Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Kranenburg, R., Lange, S., Meissner, S.: Enabling Things to Talk: Designing IoT solutions with the IoT Architectural Reference Model. ISBN 978-3-642-40403-0, Springer (2013)
4. Kim, Y.-W.: IoT Reference Architecture versus IoT Platform From Standardization Viewpoint, http://docbox.etsi.org/Workshop/2014/201407_IOTWORKSHOP/ETRI_KIM.pdf (2014)
5. Kiljander, J., D'Elia, A., Morandi, F., Hyttinen, P., Takalo-Mattila, J., Ylisaukko-Oja, A., Soininen, J.-P., Cinotti, T.S.: Semantic Interoperability Architecture for Pervasive Computing and Internet of Things. In: IEEE Open Access Journal, Volume 2 (2014)
6. Tawfik, A., Giunchiglia, F., Maltese, V.: A Collaborative Platform for Multilingual Ontology Development. Technical Report, Trento University (2014)
7. Ganbold, A., Farazi, F., Giunchiglia, F.: Managing Language Diversity across Cultures: the English-Mongolian Case Study. Technical Report, Trento University (2013)
8. Akerman, A., Tyree, J.: Using Ontology to Support Development of Software Architectures. In: IBM Systems Journal (45:4), pp. 813-825 (2006)
9. Ameller, D., Franch, X.: Ontology-Based Architectural Knowledge Representation: Structural Elements Module. In: Advanced Information Systems Engineering Workshops, Lecture Notes in Business Information Processing, Springer Berlin Heidelberg, pp. 296-301 (2011)
10. Nikitina, S.: Translating Qualitative Requirements into Design Choices: Evaluating the Method Proposed in the Architectural Reference Model for the Internet of Things. Master thesis, http://iotforum.org/wp-content/uploads/2014/10/Nikitina_S_Thesis.pdf (2014)

A Proposal of a Model using Kansei Evaluation integrated with Fuzzy Rules and Self-Organizing Map for Evaluation of Bio-Food Products

Hai Van Pham* and Khang Dinh Tran

School of Information Technology and Communication, Hanoi University of Science and Technology
No.1, Dai Co Viet St., Hanoi, Vietnam

{haipv,khangtd}@soict.hust.edu.vn

Abstract. Agricultural product intelligence is a new way for biotechnology that can be made multiple food products with a variety of characteristics, enhanced flavor and nutritional quality of foods. To evaluate food products of bio-food products for improvement of the food quality in global bio-food markets, cross-cultural customer behaviors are mostly influenced with healthy food in global markets. In this study, we propose a new approach using *Kansei* Evaluation integrated with fuzzy rules and Self-Organizing Map (SOM) model, together with customers and expert sensibilities and preferences for selection of the appropriate alternatives (bio-food and food technology), matched with mass customer and expert group behaviors. To confirm the model's performance, the proposed approach has been tested in experiments for the validated model and applied in several domains in Asian countries.

Keywords: Kansei Evaluation, Food Biotechnology, Bio-food Evaluation, fuzzy rules, Self-Organizing Maps

1 Introduction

The potential food biotechnology is a key biotechnology engineering to produce good bio-food, attractive marketing, or healthy nutrition in daily life. Food biotechnology has been widely produced healthy foods in daily life [1]. Food biotechnology is one of the ways quickly producing fresh foods and improving quality. Researchers need to determine if each potential new food product will be a useful, beneficial and safe development. Customers and consumers would be expected good food products as well as quality foods. In conventional methods [1] [2] [5] [6], most approaches have been investigated bio-food evaluation of good features based on food factory standards. However, the weaken points of

* Corresponding to Hai Van Pham, email: haipv@soict.hust.edu.vn

these approaches using group customer/expert sensibilities and preferences are not aggregated concurrently that affects to evaluate a bio-food quality in terms of customer feedbacks. In food biology evaluation, Kansei evaluation makes it possible to quantify expert and customer perception, sensation, cognition, sentiment, and impression about food quality [3]. In Kansei Evaluation, we have determined adjective pairs called Kansei words in pairs: Synonym - Antonym, Synonym - Not Synonym. For instance, the pairs of adjectives "good - bad"and "satisfied- not satisfied"are Kansei words. Emotions are part of human behavior in certain sensibilities that affect human final decisions [3]. Collaborative Decision Making (CDM) can be defined as a decision problem with the selection of alternatives, using CDM obtains dynamic decision solutions talking into account these preferences. Self-Organizing Maps (SOMs) were invented by Kohonen as a computational method for the visualization of high-dimensional data [4]. The study in this paper is to solve the existing problems in a dynamic evaluation of the appropriate alternatives (bio-food, food technology and food product) in dynamic markets. The aim of proposed approach is to evaluate alternatives (bio-food, food technology and food products), using Collaborative Decision Making (CDM), together with expert sensibilities and emotions. The advantages of the proposed approach are presented in this paper by dealing with multi-cultural customers in market dynamics as follows: 1) qualitative factors with uncertainty in dynamic market conditions, consisting of expert preferences are considered through the model; 2) the framework is used to quantify expert's sensibilities and emotions using collaborative decision making decisions. To confirm the model's performance, the proposed system has been validated by experimental results for the demonstration of this study.

2 Kansei Evaluation in quantification of human sensibility and behavior

2.1 *Kansei* word matrix construction

In the preliminary experiment, the surveys were done by 30 customers including local, international students and experts from Hanoi Fresh Vegetable and Fruit Institute with export companies for the markets. There were 5 experts and 25 customers who participated in the surveys for selection of 14 adjectives as pairs of *Kansei* words in total of 16 *Kansei* words influenced to these criteria. In the final data collection, we have selected 14 *Kansei* words of the most relevant words with redundance of 02 *Kansei* words using Factor Analysis that are influenced to these criteria for evaluation of biology-food products, as shown in Table 1.

A *Kansei* matrix is constructed using *Kansei* words for the steps as follows:

1. Collect customer surveys in the preliminary experiment using Semantic Differential (SD) method with a five-point scale definition (0: *oppose*, 0.25: *almost oppose*, 0.5: *have no preference*, 0.75: *almost agree*, 1: *agree*).
2. Consider the most appropriate *Kansei* words in the bio-food market belonging to *Kansei* criteria evaluation.

3. Structure *Kansei* words using Factor Analysis to identify conditional factor loadings and commonalities in the model.
4. Select the most appropriate *Kansei* words and construct a *Kansei* matrix from the average of *Kansei* weights.
5. Update *Kansei* weights to the *Kansei* matrix in a Database.

Table 1. Pairs of *Kansei* words for *Kansei* evaluation

No	Positive word	Negative word	Factor	Criteria Evaluation
1	Good	Not good	Product name	Product Information
2	Pleasant	Unpleasant	Bio-food product	
3	Famous	Not famous	Brand name	
4	Flavor	Not flavor	Customer satisfactory	
5	Cheap	Expensive	Bio-food Price	
6	Satisfactory	Not satisfactory	Product quality	Product quality
7	Low	High	Customer satisfactory	
8	Acceptable	Not acceptable	International standard	
9	Satisfactory	Unsatisfactory	Customer behavior	
10	Global	Local	International market	Markets
11	Stable	Changing	Asian market	
12	Standing	Falling	Government market	
13	Liked	Disliked	Asian culture	Culture
14	Preferable	Not preferable	Vietnamese culture	

2.2 *Kansei* matrix evaluation

Let $X^S = \{X_1^S, X_2^S, ..., X_m^S\}$ be a set of *Kansei* words that use to evaluate bio-food products, where m is the number of *Kansei* words. In order to quantify customer's sensibilities in evaluating bio-food products, we have refined for the most important *Kansei* words in X^S to evaluate a bio-food product with respect to criteria in stock market S.

Let $W_m^S = \{W_m^-, W_m^+\}$ be opposite pairs of *Kansei* words with respect to X_m^S. Let P customers collect their preferences by surveys. To evaluate bio-food product C_j^S, its *Kansei* weight w_{ij}^t represents by the *i-th Kansei* word of the *j-th* bio-food product evaluated by the *t-th* expert. Hence, the average weight of *i-th Kanse*i word is evaluated by P customers, as given by Eq.(1).

$$k_{ij}^S = \frac{1}{P} \sum_{t=1}^{P} w_{ij}^t \tag{1}$$

$K^S_{n\times m} = (k^S_{ij})_{(n\times m)}$ is a *Kansei* matrix construction, where n and m is the number of bio-food products and *Kansei* words respectively. Table 2 illustrates a sample of the *Kansei* matrix consisting of alternatives in *Kansei* evaluation.

Table 2. The *Kansei* score matrix in an evaluation

ID No	X^S_1	X^S_2	**...**	X^S_{16}	**Bio-food product**
1	k_{11}	k_{12}	**...**	k_{1m}	P_1
...	**...**	**...**	**...**	**...**	**...**
n	k_{n1}	k_{n2}	**...**	k_{nm}	P_n

3 Proposed approach

3.1 Human emotions in decision making

Human emotions in decision making can be presented as an integration of logical rules, quantitative knowledge and reasoning evidence. Figure 1 shows the quantification of expert sensibilities and emotions.

In common sense human reasoning, linguistic expressions represent rules for expert decision situations. To quantify emotions and sensibilities in reasoning of expert in dynamic market environments, we use fuzzy rules as illustrations of an example as follows:
Rule 1: IF Japanese people prefer an organic apple juice with its red color AND satisfaction with its fresh fruit THEN The system marks positive emotion status liked++
Rule 2: IF Chinese people do not like an organic apple juice AND Chinese markets have many fresh apples with these low prices THEN The system marks negative emotion status disliked- -
Note that emotion expert status represents in the subset disliked- -, disliked-, neutral, satisfied+, satisfied++, as described human reasoning emotion decisions.

Common Sense Human Reasoning can be presented as an integration of fuzzy rules, quantitative knowledge and reasoning evidence. Linguistic expressions can be used to represent rules for expert decision situations. To quantify the Common Sense Human Reasoning of expert e_i in dynamic market environments, we use the following logical rules as **Rule *i*** can be expressed by Eq.(1):

IF ***Condition* 1** AND...AND ***Condition m***
THEN ***Actions***

Note that expert decision status is represented in a five step scale {invest++, invest+, neutral, risk-, risk- -}

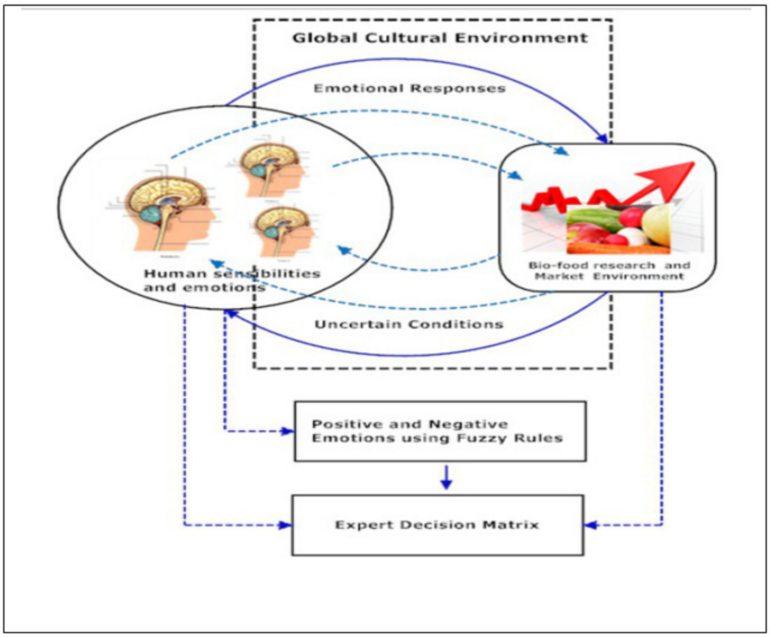


Fig. 1. An Overview of Human emotions in decision making

4 Steps in the Proposed Model

The proposed model aims to explain how we evaluate bio-food products in terms of expert preferences and customer behaviors. This uses uncertainty of quantitative and qualitative factor weights, together with the quantification of customer sensibilities in bio-food products environments. These weights can be transformed through the framework, representing in interval values [0,1]. Mechanisms of data process in the proposed model are divided into four steps as follows:

Step 1. Kansei Evaluation to quantify expert sensibilities and preferences. Pairs of adjectives called Kansei words are collected from bio-food markets. Based on the experts' experiences, Semantic Differentials (SD) method is applied to refine pairwised Kansei words. All refined-adjective words use to evaluate alternatives based on expert preferences. In Kansei evaluation, expert sensibilities and emotions are quantified in weights, representing in an internal value [0,1] with factor weights and stored data sets in a Database.

Step 2. Collaborative Decision Making using sensibilities and preferences in Bio-food Evaluation. Experts surveys are collected that divided into several groups. Each group of experts makes surveys and provides its preferences using collaborative decision making.

Step 3. SOM visualization by updating weights of Kansei data sets together with expert sensibilities and emotions. Kansei bio-food evaluation matrix, as discussed in Section 2.2 is divided into two sub matrices. Kansei Bio-food matrix and Kansei decision matrix are the same as bio-food alternatives, indicators and Kansei words. The Expert decision matrix using collaborative decision making of expert groups by calculating weights. These matrices are visualized by SOM to aggregate customer/expert preferences.

Step 4. Selection of alternatives matching with customer/ expert preferences. The outcomes of the systems are shown in the best bio-food products, matched with customer/expert preferences using collaborative decision making. To select the quantity of bio-foods, we apply logical fuzzy rules to consider the bio-food results by dealing with group customer/expert decisions as follows:

Step 4.1 Calculation of decision maker preference distances with updating *Kansei* bio-food matrix. In order to aggregate all decision maker preferences, the decision maker preference distance $d^S_{e_i \to e_j}$ between two vectors $D^S_{e_i}$ and $D^S_{e_j}$ represents by decision maker preferences, in food market S as defined by Euclidean distance given by Eq.(2).

$$d^S_{e_i \to e_j} = \parallel D^S_{e_i} - D^S_{e_j} \parallel \tag{2}$$

Step 4.2. Calculating weights of decision maker preference distances. To select bio-food products $p^S_t|(t = 1, ..., c)$ in food market S, the decision maker preference distance $d^S_{e_i \to e_j}$ is represented by m^t_{ij} calculated from the *Kansei* product attribute distance v^t_{ij} evaluated by decision maker e^S_i of his/her group at iteration t and the *Kansei* bio-food weight w^t_{ij} of the decision maker group. The weight of m^t_{ij} is expressed by Eq.(3).

$$m^t_{ij} = \parallel \frac{1}{P} \sum_{j=1}^{k} w^t_{ij} - v^t_{ij} \parallel \tag{3}$$

where (i = 1,...,q, j = 1,...,k) and p is the number of decision makers in each group.

Step 4.3. Updating *Kansei* bio-food weights. A Decision matrix $A^S_{q \times k}$ is updated by its weights given by Eq.(4). After that, the Decision matrix $A^S_{q \times k}$ is joined with *Kansei* bio-food matrix $M^S_{n \times p}$ and its weights are updated to $M^S_{n \times p}$.

$$m^{t+1}_{ij} = m^t_{ij} + \beta^S_j (\parallel \frac{1}{P} \sum_{j=1}^{m} w^t_{ij} - v^t_{ij} \parallel) \tag{4}$$

where β^S_j is a set of decision maker preferences as defined in a five-point scale (0: *oppose*, 0.25: *almost oppose*, 0.5: *have no preference*, 0.75: *almost agree*, 1: *agree*)

To select c bio-food product $(p_1^S, p_2^S, ..., p_c^S)$, the similar steps are repeated between Step 4.1 and Step 4.3 until c decision maker groups with updated weights to the Decision matrix completely.

5 Results and Discussions

The proposed model has been employed several domains of Asian markets for demonstration of the proposed approach. In data collection, web-based application is designed to allow any authorized user via the Intranet / Internet connection. Figure 2 shows an example of results for the illustration of collaborative decision making of expert and customer groups for evaluation of bio-food products on the Asian markets.

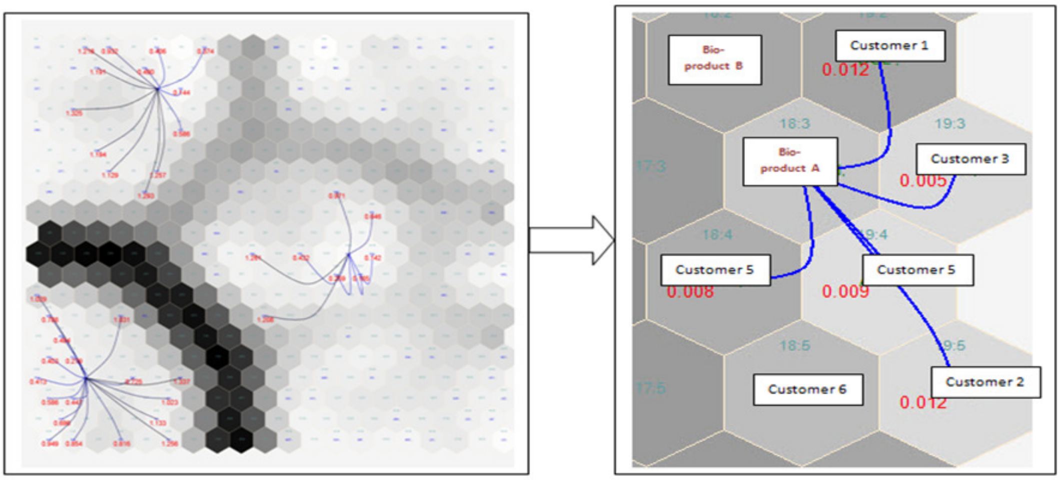


Fig. 2. The results of customer satisfaction with bio products on maps in detail

The proposed model is used to evaluate dynamic solutions of bio-food technology, based on collaborative expert preferences and customer behaviors. The simulation results showed that customer behaviors, together with expert preferences matched with bio-food products. Decision makers can dynamically evaluate bio-food properties on map results as well as optimal decisions. Compared to conventional methods in evaluating bio-food products, most approaches have been used statistic methods in evaluation, the proposed model has been figured out dynamic evaluation in quantification of customer sensibility and behaviour. The new approach using Kansei evaluation is to quantify human sensibilities and emotions about bio-food quality in market and bio-food research environments. This approach is also illustrated with a case study of experimental results to validate the model. In future works, the proposed approach is extended to evaluate multiple food products in daily life.

6 Conclusion

The proposed approach has been applied in evaluation of bio-food products, based on collaborative expert preferences and customer behaviors. The simulation results showed that customer behaviors, together with expert preferences matched with bio-food products. Decision makers can dynamically evaluate bio-food properties on map results as well as optimal decisions. The new approach using Kansei evaluation is to quantify expert sensibilities and emotions about bio-food quality in market and bio-food research environments. This approach is also illustrated with a case study of experimental results to validate the model. In future works, the proposed approach is extended to evaluate multiple food products in daily life.

References

1. Nolwennig R., Jean R., and Gard D, 2009, Information effects on consumer attitudes toward three food technologies: Organic production, biotechnology, and irradiation, Food Quality and Preference (20) 586-596.
2. Rita Batista, Maria Margarida Oliveira, 2009, Facts and fiction of genetically engineered food, Trends in Biotechnology (27) 277-286.
3. Salem, B. and Nakatsu, R. and Rauterberg, M., 2005, Kansei Experience: Aesthetic, Emotions and Inner Balance, International Journal of Cognitive Informatics and Natural Intelligence (3) 54-64.
4. T. Kohonen,(1995), Sefl-Organizing Maps, Spiringer-verlag.
5. Bologna, and Italy, 2013, Bio-based technologies in the context of European Food Innovation Systems: Outcomes from the EFFoST Annual Meeting, Trends in Food Science and Technology, (37) 162-165
6. P.M. Davidson, H. Bozkurt Cekmer, E.A. Monu, C. Techathuvanan, 2015, The use of natural antimicrobials in food: An overview, Handbook of Natural Antimicrobials for Food Safety and Quality, (5) 1-27

Improved algorithm of unsatisfiability-based Maximum Satisfiability

Yiyuan Wang, Dantong Ouyang, Liming Zhang*

Symbol Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China
College of Computer Science and Technology, Jilin University, Changchun 130012, China

Abstract. Recently, the Maximum Satisfiability (MaxSAT) discovers a rapidly increasing number of practical applications in a wide range of different areas. Unsatisfiability-based MaxSAT algorithms have been put forward aimed at improving the performance of solving MaxSAT problem. In these algorithms, several excellent Boolean Satisfiability (SAT) solvers are employed to iteratively identify unsatisfiable sub-formulas and the additional blocking variables are also applied to relax some of initial clauses or encode Boolean constraints into a CNF formula. Firstly, this paper proposes two novel optimizations to reduce the search space: finding all disjoint unsatisfiable cores with the original clauses removed and no blocking variables added; avoiding generating the largest number of blocking variables' clauses with heuristic strategy. Moreover, in order to reduce the number of satisfying iterations, the partial assignment is introduced into satisfiability-based framework as the tight bound accelerator which could find a smaller number of blocking variables. Experimental results show that the unsatisfiability-based algorithm with these optimizations results in an improved and more effective solver for MaxSAT problem than previous algorithms. In addition,this results in consistent performance gains in most cases and on average 1.5 times speed-up for MaxSAT with state-of-the-art algorithms.

Introduction

The classical NP-complete problem of Boolean Satisfiability (SAT) [8] has been a growth of interest in not just the theoretical computer science community, but also in various areas where feasible solutions to these problems are significantly important in many practical applications. These practical applications have been able to successfully apply SAT as a decision procedure to determine whether these instances are SAT or UNSAT. However, there are more extra demands of the SAT problem that exceed this decision procedure of SAT solvers, such as Maximum Satisfiability problem (MaxSAT) [12, 3] and Pseudo-Boolean Optimization (PBO). Moreover, using the strong relationship between these problems, several new algorithms have been developed [21].

MaxSAT as a well-known problem in Computer Science, consists of identifying the largest number of clauses in a CNF formula that can be satisfied. The recent application of MaxSAT and variants in design debugging and verification of complex designs [4, 10, 20] motivated the development of new MaxSAT algorithms, capable of solving large structured problem instances common to these application domains. Many exact methods for finding MaxSAT have been developed in recent years. Davis-Putnam-Logemann-Loveland procedure [5] explores a binary search tree and then establishes incrementally truth assignments. The result of several methods based on this procedure is very good, but they are not suitable for MaxSAT practical problems. Some other exact methods generally based on Branch and Bound (B&B) algorithms [2] have been designed to handle MaxSAT, but are unable to solve majority of problem instances. Recently algorithms based on the identification of unsatisfiable sub-formulas [7, 18, 19, 15], with blocking variables introduced into these algorithms and Pseudo Boolean Optimization used to simplify CNF, have also been developed, and are now able to tackle all these Boolean optimization problems and most of unsatisfiable instances.

This paper firstly reviews previous MaxSAT algorithm based on unsatisfiable sub-formulas named $msu4$ [19], and then proposes an improved MaxSAT algorithm based on the method of binary search [11]. Despite the improved algorithm building on $msu4$ based on unsatisfiable sub-formulas and binary search, this paper proposes one improvement as the tight bound accelerator and extra two optimizations to reduce the search space, which are novel, including: (1) reducing the number of SAT solver iterations under the control of partial assignment; (2) developing the heuristic strategy to avoid generating the largest number of blocking clauses, so as to slow down the efficiency of algorithm, when the middle value is half number of blocking variables. (3) speeding up to find all disjoint unsatisfiable cores without blocking variables added into these clauses in the early iterative execution of binary search's preprocess. The new improved algorithm also uses several techniques [6] to utilize BDDs for encoding cardinality constraints as Boolean circuits. Experimental results show that the new MaxSAT algorithm is observably faster than the previous MaxSAT algorithms in a wide range of unsatisfiable industrial instances, demonstrating the robustness of the proposed approach.

Preliminaries

Firstly a finite set of Boolean variables is assumed X=$\{x_1,x_2,x_3,\ldots,x_n\}$. A CNF formula φ is defined on the conjunction of clauses w_i and each clause consists of the disjunction of literals, which is either a variable x_i or its complement $\overline{x}_i$. In the context of search algorithms of SAT, variables can be assigned a logic value, either 0 or 1. Alternatively, variables can defined as a function v: $X \rightarrow \{0,u,1\}$, where u denotes a variable whose value is uncertain. When all of variables are assigned a value in $\{0,1\}$, then v is referred to as a complete assignment. Otherwise it is a partial assignment [17] where some variables do not have values and are assigned u, which will be used in the following improved algorithm.

In this section Maximal Satisfiability and Minimal Unsatisfiability are firstly introduced. Afterwards, the previous MaxSAT algorithm using unsatisfiable cores named $msu4$ is described.

Maximal Satisfiability and Minimal Unsatisfiability

Our improved algorithm and previous methods are based on maximal satisfiability and minimal unsatisfiability. The MaxSAT problem as an optimization problem corresponds to the minimization of the number of false clauses in a CNF formula, which means that MaxSAT problem discovers an assignment to the variables of a CNF formula that maximizes the number of satisfied clauses.

In order to illustrate this problem, Maximally Satisfiable Subset (MSS) is defined as a subset of clauses in a CNF formula, which is satisfiable and then adds any of the other clauses in an original formula to MSS leading new clauses set to unsatisfiable. In this context, Minimal Unsatisfiable Subset (MUS) is an inconsistent subset of the clauses of the original formula and dropping one of its clauses will make it satisfiable. MSS and MUS [14, 7] correspond to the following rules:

$$MSS(\varphi) = \begin{cases} m \quad is \quad satisfiable, & \text{if m}\subseteq\varphi \\ m \cup \{a\} \quad is \quad unsatisfiable, & \forall\text{a}\in(\varphi\text{-m}) \end{cases}$$

$$MUS(\varphi) = \begin{cases} m \quad is \quad unsatisfiable, & \text{if m}\subseteq\varphi \\ m - \{a\} \quad is \quad satisfiable, & \forall\text{a}\in\text{m} \end{cases}$$

Notice that MSS as a satisfied subset of clauses cannot become bigger and MUS as an unsatisfiable subset that removes any of its clauses can render it satisfiable. The relationship between MaxSAT and MSS is subtle. Given an unsatisfiable instance, MaxSAT is the largest subset of clauses that can be satisfiable which means to find MSS with maximal number. However, all MSSes may have different sizes and cannot be al-l MaxSAT solutions with maximal number. Removing the clauses that are not included in the MSS makes CNF formula satisfiable. Therefore Minimal Correction Set (MCS) is introduced to describe the complement of MSS and the set of MCSes is shown:

$$MCSes(\varphi) = \{m | m \subseteq \varphi \quad and \quad (\varphi - m) \in MSSes(\varphi)\}$$

MCS as the complementary point of MSSes offers the real link to MUSes. Note that any clause of MUS in a CNF formula makes the formula unsatisfiable and the complement formula satisfiable,thus every MSS must contain at least one of clauses of every MUS. This relationship between MCSes and MUSes can usually be described as a solution to a set converting problem. Specifically, each hitting set of the MUSes is an MCS. Given a collection of sets, hitting set intersects all sets in the collection in at least one element and the problem of hitting set is an equivalent reformulation of set covering. A MCS is a hitting set of the MUSes with the additional limit that it cannot become smaller without losing its defining property: it is an irreducible hitting set and each hitting set of MCSes is an MUS. The example formula is shown to explain their relationship between MSSes, MCSes and MUSes.

$$\varphi = (x_1) \wedge (\overline{x}_1 \vee \overline{x}_2) \wedge (x_2) \wedge (\overline{x}_2)$$

$$MSSes(\varphi) = \big\{\{x_1, \overline{x}_1 \vee \overline{x}_2, \overline{x}_2\}, \{\overline{x}_1 \vee \overline{x}_2, x_2\}, \{x_1, x_2\}\big\}$$

$$MCSes(\varphi) = \big\{\{x_2\}, \{x_1, \overline{x}_2\}, \{\overline{x}_1 \vee \overline{x}_2, \overline{x}_2\}\big\}$$

$$MUSes(\varphi) = \big\{\{x_1, \overline{x}_1 \vee \overline{x}_2, x_2\}, \{x_2, \overline{x}_2\}\big\}$$

In the example above, every MUS of a CNF formula is an irreducible hitting set of the MCSes of that CNF formula. In the given CNF formula above, the number of maximal satisfiable clauses is 3. More importantly, MSS and MUS also can tell us the relationship between MUS and MaxSAT problem.

A MaxSAT algorithm using unsatisfiable cores

Recently, several methods based on unsatisfiable cores are proposed. $Msu4$ which applies improved blocking variable into clauses is more effective than other algorithms in the experiment results. More importantly, $msu4$ avoids interacting with a PBO solver and instead is a fully SAT-based solver that relies on the most effective techniques, such as Boolean Constraint Propagation, conflict based learning and conflict-directed backtracking [16]. Finally, the two advantages of $msu4$ lie in adding at most one blocking variable into each clause in a CNF formula which can avoid quickly increasing the number of blocking variables and clauses, and applying linear programming encoding of pseudo boolean constraints with BDDs and sorting networks.

Proposition 1 ***MaxSAT Upper Bound(UB)****: In a CNF formula, φ corresponds to the number of clauses. Let K denote disjoint unsatisfiable cores, then φ-K corresponds to an MaxSAT Upper Bound.*

Proposition 2 ***MaxSAT Lower Bound(LB)****: Let B define as the set of blocking variables assigned value TRUE when clauses become satisfiable, then φ-B denotes a MaxSAT Lower Bound.*

In previous subsection, some relationships between MUS and MaxSAT are briefly introduced and these properties are employed in the algorithm to define MaxSAT Upper Bound and Lower Bound [19]. Some details for $msu4$ are shown in [19].

Improved MaxSAT algorithm with some strategies

As shown previously, unsatisfiable-based algorithm is able to tackle MaxSAT problem. However, $msu4$ works by the redundant lower and upper bounds in some SAT solver iterations. This section proposes an improved algorithm based on $msu4$ with two key optimizations, which are used as the tight bound accelerator. Firstly the thought of binary search [11] (its algorithm named BIN) can be imported into $msu4$ for clearly improving its efficiency by decreasing the redundant calculation. Secondly, based on the idea of binary search, this paper proposes some key optimizations: (1)the use of partial assignment computing a smaller number of blocking variables assigned value

TRUE instead of the middle value of binary search in each satisfying assignment. (2)the heuristic strategy HS_{BS} to avoid generating the largest number of clauses containing blocking variables when the middle value is nearly half number of blocking variables. (3)acceleration to find disjoint unsatisfiable cores by iteratively computing the original formula and temporarily deleting these cores that have already been found.

Unsatisfiable core-based MaxSAT algorithm using binary search

In algorithm 1, compared with the upper bound and lower bound in the $msu4$, two new bounds, which denote the lower and upper bounds of lower bound, are firstly proposed and then used into algorithm 1 and algorithm 2.

Proposition 3 ***Lower Bound of Lower Bound (L_LB)****: φ corresponds to a CNF formula. Remove any L_LB clauses to render new φ^* unsatisfiable. In our algorithm 2, L_LB is initialized to the number of unsatisfiable cores.*

Proposition 4 ***Upper Bound of Lower Bound (L_UB)****:φ corresponds to a CNF formula. Remove the given L_UB clauses to make new φ^* satisfiable. In algorithm 2, L_UB is firstly assigned to the number of blocking variables.*

Proposition 5 *[11] Binary search for MaxSAT executes $\Theta(log(W))$ calls to a SAT oracle in the worst case.*

Compared with $msu4$, algorithm 1 uses the method of binary search to acquire MaxSAT solution and is also based on Proposition 5 defined in the BIN to guide algorithm 1. The pseudo code for $msu5\_bin$ is shown in Algorithm 1.

In $msu4$, lower bound only depends on the value of blocking variables when φ is satisfiable, and its variation is passive. Considering that $msu4$ calculates the size of lower bound by inches, this paper presents the binary search, whose algorithm in computer science finds the position of a specified value within a sorted key. Algorithm 1 employs the binary search to take the initiative to alter two bounds in order to accelerate the growth of two bounds for reducing the number of SAT solver iterations, which will solve the MaxSAT problem quickly.

In algorithm 1, there are several differences between $msu4$ and $msu5\_bin$. L_UB is initialized to the number of initial clauses and L_LB is to 0. Before each call to the SAT solver, BS_B which corresponds to the middle variable, needs to be computed under the method of binary search (line 6). Afterwards $CNF(\sum_{i \in V_B} b_i \leq BS_B)$ is generated and then added to φ_w (line 7 and 8). L_LB is only updated on iterations when the SAT solver returns false. Considering such an iteration, for which the unsatisfiable core does not include initial clauses (line 17), L_LB will be assigned to BS_B. When the unsatisfiable core contains original clauses $I > 0$, this process is the same as $msu4$ from line 13 to 16. Consider now a satisfiable iteration. It occurs when the number of blocking variables assigned 1 is sufficient to make the current CNF formula get a satisfiable assignment, L_UB will be updated(line 21). But it can not guarantee that this number is minimal. When $L\_LB + 1 \geq L\_UB$, it is an answer of the MaxSAT solution.

Algorithm 1 Improving MaxSAT Algorithm based on binary search

▷ the running condition of algorithm 1 is that φ is unsatisfiable
$L\_LB \leftarrow 0$
$L\_UB \leftarrow |\varphi|$
$\varphi_w \leftarrow \varphi$
while TRUE do
  $BS_B \leftarrow (L\_LB + L\_UB)/2$ ▷ BS_B is the middle variable in the binary search
  $\varphi_{CH} \leftarrow CNF(\sum_{i \in V_B} b_i \leq BS_B)$
  $(st, \varphi_c) \leftarrow SAT(\varphi_w \cup \varphi_{CH})$
  if st = **UNSAT then**
    $\varphi_I \leftarrow \varphi_c \cap \varphi$
    $I \leftarrow \{i | w_i \in \varphi_I\}$
    $V_B \leftarrow V_B \cup I$
    if $|I| > 0$ **then**
      $\varphi_N \leftarrow \{w_i \cup \{b_i\} | w_i \in \varphi_I\}$
      $\varphi_a \leftarrow CNF(\sum_{i \in I} b_i \geq 1)$
      $\varphi_w \leftarrow (\varphi_w - \varphi_I) \cup \varphi_N \cup \varphi_a$
    else
      $L\_LB \leftarrow BS_B$
    end if
  else
    $L\_UB \leftarrow BS_B$
  end if
  if $L\_LB + 1 \geq L\_UB$ **then**
    return a satisfiable assignment
  end if
end while

Improved algorithm based on binary search

Although the performance of $msu5\_bin$ is superior to previous methods, three novel improvements are proposed in algorithm 2, which can be more efficient to optimize the MaxSAT algorithm.

Firstly, when the SAT solver returns satisfiable, the upper bound only relies on the middle variable in algorithm 1, which may not be the minimal number of blocking variables assigned 1. In detail, after some satisfiable iterations, the assignments of upper bounds are usually redundant. Our algorithm 2 (named $msu6\_bd$) proposes the technology of partial assignment, which could effectively reduce the number of satisfying iterations in algorithm 1. Comparing the number of blocking variables (assigned value TRUE) in partial assignment with the value of middle variable, the smaller value will be assigned to the upper bound. Specifically, a partial assignment might not include some blocking variables called irrelevant variables, which means that SAT solver only needs to analyze several blocking variables and discard some other blocking variables if they are not used for satisfying any clause. Our Algorithm is built on the interface of PicoSAT [1] to eliminate unnecessary SAT tests by simplifying satisfying assignments using standard partial assignment.

Algorithm 2 Our Improved MaxSAT Algorithm based on binary search

▷ the running condition of algorithm 2 is that φ is unsatisfiable
$IN_C \leftarrow \phi; UN_C \leftarrow \phi; CS_C \leftarrow \phi; V_B \leftarrow \phi$
while $st \neq SAT$ **do**
    $(st, \varphi_c) \leftarrow increSAT(\varphi - IN_C)$
    if st = **UNSAT** **then**
        $V_B \leftarrow V_B \cup \{i | w_i \in \varphi_c\}$
        $IN_C \leftarrow IN_C \cup \varphi_c$
        $UN_C \leftarrow UN_C \cup \{w_i \cup \{b_i\} | w_i \in \varphi_c\}$
        $CS_C \leftarrow CS_C \cup CNF(\sum_{i \in \varphi_c} b_i \geq 1)$
    end if
end while
$\varphi_w \leftarrow (\varphi - IN_C) \cup UN_C \cup CS_C$
$L\_LB \leftarrow |IN_C|; L\_UB \leftarrow |V_B|$
while TRUE do
    $BS_B \leftarrow (HS_{BS}(L\_LB))?(L\_LB + L\_UB)/2 : (3L\_LB + L\_UB)/4$ ▷ heuristic strategy reduces the size of CNF formula
    $\varphi_{CH} \leftarrow CNF(\sum_{i \in V_B} b_i \leq BS_B)$
    $(st, \varphi_c) \leftarrow increSAT(\varphi_w \cup \varphi_{CH})$
    if st = **UNSAT** **then**
        $\varphi_I \leftarrow \varphi_c \cap \varphi$
        $I \leftarrow \{i | w_i \in \varphi_I\}$
        $V_B \leftarrow V_B \cup I$
        if $|I| > 0$ **then**
            $\varphi_N \leftarrow \{w_i \cup \{b_i\} | w_i \in \varphi_I\}$
            $\varphi_a \leftarrow CNF(\sum_{i \in I} b_i \geq 1)$
            $\varphi_w \leftarrow (\varphi_w - \varphi_I) \cup \varphi_N \cup \varphi_a$
        else
            $L\_LB \leftarrow BS_B$
        end if
    else
        $V_S \leftarrow$|the number of blocking variables assigned value 1|
        $L\_UB \leftarrow (V_S < BS_B)?V_S : BS_B$ ▷SAT solver selects partial assignment
    end if
    if $L\_LB + 1 \geq L\_UB$ **then**
        return a satisfiable assignment
    end if
end while

Secondly, when the middle variable in algorithm 1 is assigned to half number of blocking variables, it could generate the largest number of clauses for pseudo boolean constraints. Considering that $x_1 + x_2 + \ldots + x_n \geq k$ results in BDDs with $(n - k + 1) \times k$ nodes [6], the number of BDD nodes is maximum when $k = n/2$. Based on above, algorithm 2 proposes the heuristic strategy HS_{BS} to avoid this situation in binary search. The heuristic HS_{BS} function is, when BS_B is approximately equal to $|V_B|/2$, BS_B is assigned to $(3L\_LB + L\_UB)/4$ which is nearly equal to $|V_B|/4$.

Lastly, when algorithm 1 iteratively obtains an unsatisfiable core each time, the CNF formula will be added more and more clauses generated by blocking variables. For this reason, algorithm 2 starts with finding all disjoint unsatisfiable cores as a prepossessing under the smaller search space with original clauses removed and no blocking variables added. In detail, whenever algorithm 2 finds an unsatisfiable core, some clauses in this core will be temporarily removed. When SAT solver can no longer obtain any unsatisfiable core, the CNF formula will be restored and then the method of binary search continues to solve MaxSAT problem. For this unsatisfiable core-solving preprocessing that requires to iterative calls to a SAT solver, the well-known incremental interface of PicoSAT solver can be considered, which is a SAT solver that after running a CNF formula, keeps all clauses and the internal state that can be reused when inserting additional clauses or deleting information learned from clauses. The pseudo code for $msu6\_bd$ is shown in Algorithm 2.

Algorithm 2 begins with finding out all disjoint unsatisfiable cores from line 2 to 11. Afterwards φ_w is set as $(\varphi - IN_C) \cup UN_C \cup CS_C$, which shows that φ removes unsatisfiable cores, and then adds those clauses including blocking variables and pseudo boolean constraints (line 12). L_LB will be initialized to the number of unsatisfiable cores and L_UB is assigned to the number of blocking variables, aiming at reducing the number of SAT solver iterations and speeding up to enhance the performance of algorithm 1 (line 13). In the process of binary search, the function of HS_{BS} is to prevent the number of blocking clauses becoming large when HS_{BS} is nearly equal to $V_B/2$ (line 15). More importantly, L_UB is assigned to the smaller value between V_S and BS_B (line 31).

Experiment Results

This section presents the experimental results of the improved MaxSAT algorithm ($msu6\_bd$). In order to evaluate this new method, *pbo* [6], $msu4$ [19] and $msu5\_bin$ which is built on the $msu4$ using the binary search [11] are considered as the comparison algorithms. The framework of our algorithm is implemented using a state-of-the-art SAT solver named PicoSAT-V953 [1].

In order to evaluate the improved MaxSAT algorithm, a set of instances are selected. These instances are obtained from existing unsatisfiable subsets of crafted and industrial MaxSAT benchmarks, the SAT competition archives and SATLIB. The majority of the considered industrial instances are originally from EDA applications, including model checking, equivalence checking , and test pattern generation. The total number of unsatisfiable instances considered is 447. All experiments are running on a Core 2 Dual 2.13 Ghz workstation with 4 GB of RAM and the CPU timeout of 1000 seconds for each instance.

Table 1 shows the number of solved instances for all selected benchmark sets. The improvements from $msu4$ and $msu5\_bin$ to $msu6\_bd$ are very clear. The overall number of solved instances ($msu6\_bd$) is vastly improved as it now solves more instances than $msu5\_bin$, $msu4$ and *pbo*. When solving MaxSAT crafted instances, the performances of $msu5\_bin$ and $msu6\_bd$ are parallel. Nevertheless, $msu6\_bd$ is not effective enough to solve all the selected instances. The MaxSAT algorithm based on the unsat-

Table 1. Number of Solved Instances

Ben. set	#I	pbo	msu4	msu5_bin	msu6_bd
ms_ind	321	111	178	230	**288**
ms_cra	126	24	68	**91**	90
Total	447	135	246	321	**378**

isfiable cores needs to use SAT solver to iteratively execute, which is not suitable for the time-out instances.

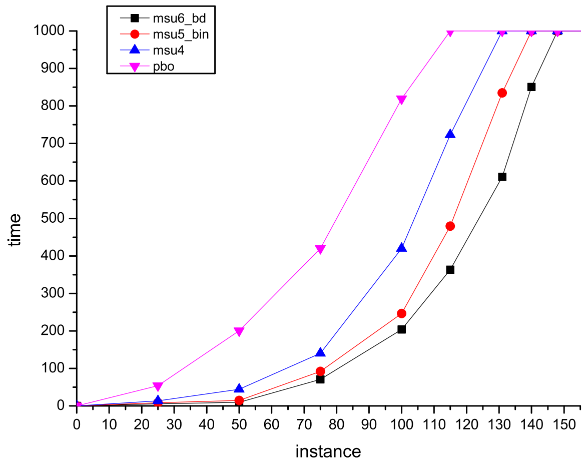


Fig. 1. Run times of algorithms

Figure 1 shows a plot by increasing run times of the test instances for each algorithm. Firstly some instances in the time limits are selected and sorted by the computing time increasingly. Clearly, pbo is inefficient. For most of test cases, it can hardly figure out the answer. In additional, $msu6\_bd$ increases in a slower rate than $msu4$ and $msu5\_bin$. It is beneficial because $msu6\_bd$ uses the binary search and three additional optimization strategies: speeding up to find all disjoint unsatisfiable cores in reduced search space, using the tight bound accelerator to reduce the number of SAT solver iterations, and applying HS_{BS} to reduce the search space of binary search.

Figure 2 plots the run times of our approach $msu6\_bd$ versus the previous algorithm $msu4$, along with the 1x, 2x, 3x and 10x lines, clearly showing the superiority of the proposed method. The time of all selected instances in two algorithms is in time limits. As can be observed, $msu6\_bd$ is clearly faster than $msu4$. $msu4$ has already proved

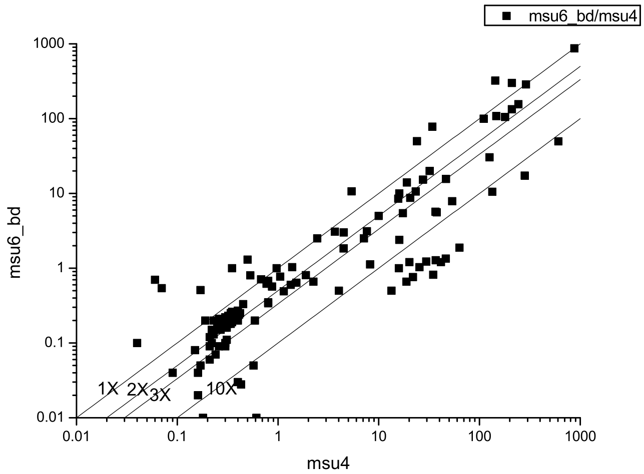


Fig. 2. Scatter plot: msu6_bd vs. msu4

that it is more effective than pbo. Despite the performance advantage of the versions of $msu6\_bd$, there are some exceptions.

Conclusions

This paper proposes an improved unsatisfiable-based MaxSAT algorithm. At first partial assignment can be employed into binary search-based framework as the tight bound accelerator: partial assignment could make upper bound select a smaller number of clauses than previous algorithms. Moreover, HS_{BS} avoids generating the largest number of clauses for pseudo boolean constraints, which can reduce the search space of binary search. Finally, another improvement strategy as the preprocessing of binary search can find as much disjoint unsatisfiable cores as possible to speed up the execution speed of MaxSAT algorithm. With the original clauses removed and no blocking clauses added, this optimization can significantly reduce search space in these number of unsatisfiable iterations. Preliminary experimental results show that these techniques significantly improve the performance of our unsatisfiability-based algorithm in solving industrial instances of MaxSAT problem. As a result, our algorithm maintains its competitive advantage over other algorithms for MaxSAT problem. Despite the promising results, additional application to $msu6\_bd$ is expected, which is to exploit structure information [9, 13] to accelerate MaxSAT algorithm in design debugging.

Acknowledgment

The authors are thankful to the editor and anonymous reviewers for their valuable comments that help us improve the quality of the paper.

References

1. Biere, A.: Picosat essentials. Journal on Satisfiability, Boolean Modeling and Computation 4(75-97), 45 (2008)
2. Borchers, B., Furman, J.: A two-phase exact algorithm for max-sat and weighted max-sat problems. Journal of Combinatorial Optimization 2(4), 299–306 (1998)
3. Cai, S., Luo, C., Thornton, J., Su, K.: Tailoring local search for partial maxsat. In: Twenty-Eighth AAAI Conference on Artificial Intelligence (2014)
4. Chen, Y., Safarpour, S., Marques-Silva, J., Veneris, A.: Automated design debugging with maximum satisfiability. Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on 29(11), 1804–1817 (2010)
5. Davis, M., Logemann, G., Loveland, D.: A machine program for theorem-proving. Communications of the ACM 5(7), 394–397 (1962)
6. Eén, N., Sörensson, N.: Translating pseudo-boolean constraints into sat. Journal on Satisfiability, Boolean Modeling and Computation 2(3-4), 1–25 (2006)
7. Fu, Z., Malik, S.: On solving the partial max-sat problem. Theory and Applications of Satisfiability Testing-SAT 2006 pp. 252–265 (2006)
8. Gary, M., Johnson, D.: Computers and Intractability: A Guide to the Theory of NP-completeness. WH Freeman and Company, New York (1979)
9. Guo, T., Li, Z., Guo, R., Zhu, X.: Large scale diagnosis using associations between system outputs and components. In: Twenty-Fifth AAAI Conference on Artificial Intelligence (2011)
10. Heras, F., Larrosa, J., Oliveras, A.: Minimaxsat: A new weighted max-sat solver. Theory and Applications of Satisfiability Testing–SAT 2007 pp. 41–55 (2007)
11. Heras, F., Morgado, A., Marques-Silva, J.: Core-guided binary search algorithms for maximum satisfiability. In: Proceedings of the AAAI National Conference (AAAI). pp. 36–41 (2011)
12. Johnson, D.: Approximation algorithms for combinatorial problems. Journal of computer and system sciences 9(3), 256–278 (1974)
13. Le, B., Mangassarian, H., Keng, B., Veneris, A.: Non-solution implications using reverse domination in a modern sat-based debugging environment. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012. pp. 629–634 (2012)
14. Liffiton, M., Sakallah, K.: On finding all minimally unsatisfiable subformulas. In: Theory and Applications of Satisfiability Testing. pp. 34–42 (2005)
15. Manquinho, V., Martins, R., Lynce, I.: Improving unsatisfiability-based algorithms for boolean optimization. Theory and Applications of Satisfiability Testing–SAT 2010 pp. 181–193 (2010)
16. Manquinho, V., Marques-Silva, J.: Search pruning techniques in sat-based branch-and-bound algorithms for the binate covering problem. Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on 21(5), 505–516 (2002)
17. Marques-Silva, J., Lynce, I., Malik, S.: Conflict-driven clause learning sat solvers. SAT Handbook pp. 131–154 (2009)
18. Marques-Silva, J., Planes, J.: On using unsatisfiability for solving maximum satisfiability. arXiv preprint arXiv:0712.1097 (2007)
19. Marques-Silva, J., Planes, J.: Algorithms for maximum satisfiability using unsatisfiable cores. In: Proceedings of the conference on Design, automation and test in Europe. pp. 408–413 (2008)
20. Zhang, L., Bacchus, F.: Maxsat heuristics for cost optimal planning. In: Proceedings of the AAAI National Conference (AAAI) (2012)
21. Zhu, Z., Li, C., Manyà, F., Argelich, J.: A new encoding from minsat into maxsat. In: Principles and Practice of Constraint Programming. pp. 455–463 (2012)

Entailment-based Axiom Pinpointing in Debugging Incoherent Terminologies

Yuxin Ye[1,2], Dantong Ouyang[1,2*], Jing Su[2]

[1] College of Computer Science and Technology, Jilin University, Changchun 130012, China
[2] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Changchun 130012, China
{yeyx,ouyd}@jlu.edu.cn
sujing11@mails.jlu.edu.cn

Abstract. One of the major problems of axiom pinpointing for incoherent terminologies is the precise positioning within the conflict axioms. In this paper we present a formal notion for the entailment-based axiom pinpointing of incoherent terminologies, where the parts of an axiom is defined by atomic entailment. Based on these concepts, we prove the one-to-many relationship between existing axiom pinpointing with the entailment-based axiom pinpointing. For its core task, calculating minimal unsatisfiable entailment, we provide algorithms for OWL DL terminologies using incremental strategy and Hitting Set Tree algorithm. The feasibility of our method is shown by case study and experiment evaluations.

Keywords: ontology debugging, description logics, pinpointing, MUPS

1 Introduction

Ontology debugging becomes a challenging task for ontology modelers since the improvement of expressivity of ontology language and ontology scale[1]. Axiom pinpointing [2] is an important mean for ontology debugging. Any approach which can detect a set of axioms in the terminology that lead to logic conflict is belong to axiom pinpointing. It can be categorized into MSSs (maximally satisfiable sub-TBox), MUPS (minimal unsatisfiable sub-TBox) and justification. For finding maximally concept-satisfiable terminologies, Meyer[3] proposes a tableau like procedure for terminologies represented in $\mathcal{ALC}$. The approach of Meyer is extended by Lam[4] to get a fine-grained axiom pinpointing for $\mathcal{ALC}$ terminologies. In addition, several methods have been proposed to calculate the MUPS. Schlobach and Cornet[5] provide complete algorithms for unfoldable $\mathcal{ALC}$-TBox based on minimization of axioms for MUPS, then Schlobach[6, 7] presents a framework for the debugging of logically contradicting terminologies. Parsia[8]

* Corresponding author: Dantong Ouyang, Email: ouyd@jlu.edu.cn

extends Schlobach[5] to more expressive DLs. Baader[2] presents automata-based algorithms for reasoning in DLs with the pinpointing formula whose minimal valuations correspond to the MUPS. From the perspective of unsatisfiability, justification is the MUPS of an unsatisfiable concept. Kalyanpur has explored the dependencies between unsatisfiable classes[9], and proposed several approaches for computing all justifications of an entailment in an OWL-DL ontology[10]. Debugging tasks in OWL ontologies are in general computationally hard, so some optimization techniques are introduced for ontology debugging such as heuristic method[11] of identifying common errors and inferences, and modularization[12] for large ontologies. On the whole, various approaches achieve the result sets of axioms responsible for an unsatisfiable concept or a incoherent terminology. Hasse[13] provides a set of criteria for comparing between different approaches related to ontology debugging directly or indirectly, and that none of the surveyed approaches is universally applicable for any application scenario. Axiom pinpointing identifies conflict axioms, but practical problems remain. It is not clear which parts of axioms lead to the conflicts. and some contradictions would be lost[14]. In this paper, we try to give other notion of axiom pinpointing for incoherent terminologies, and define algorithms for this task.

The rest of this paper is organized as follows. In section 2 we briefly introduce the drawback of MUPS. Then the formal definitions about fine-grained axiom pinpointing, and the link with axiom pinpointing are presented in section 3. Section 4 presents algorithm for calculating the minimal unsatisfiable entailment. Section 5 analyzes the fine-grained axiom pinpointing with a case study and evaluates the algorithm by experimenting with common ontologies. Finally, we conclude the paper in section 6.

2 Drawback of MUPS

Axiom pinpointing[2] has been introduced in description logics to help the user to understand the reasons why consequences hold and to remove unwanted consequences by computing minimal subsets of the terminology that have the consequence. The axiom pinpointing we discuss in this paper is MUPS[5]. It's useful for relating sets of axioms to the unsatisfiability of specific concept.

Definition 1 (MUPS[5]). *A TBox $\mathcal{T}' \subseteq \mathcal{T}$ is a minimal unsatisfiability preserving sub-TBox* (MUPS) *for C in $\mathcal{T}$ if C is unsatisfiable in $\mathcal{T}'$, and C is satisfiable in every sub-TBox $\mathcal{T}'' \subset \mathcal{T}'$. The set of all* MUPS *of C in $\mathcal{T}$ is denoted as mups$(\mathcal{T}, C)$.*

Most existing approaches can obtain the different fine-grained problematic axioms on the basis of axiom pinpointing as none of these approaches define exactly what they mean by parts of axioms. Further, some logic contradictions would be lost with axiom pinpointing since it does not point out the specific location within the axioms of the logic contradiction. Let us use an example to illustrate these limitations.

Example 1. A TBox $\mathcal{T}_1$ consists of the following axioms $(\alpha_1 - \alpha_6)$, where A and B are base concepts, $A_1, ..., A_6$ are named concepts, and r and s are roles:

$\alpha_1 : A_1 \sqsubseteq A_2 \sqcap \exists r.A_2 \sqcap A_3$

$\alpha_2 : A_2 \sqsubseteq A \sqcap B$

$\alpha_3 : A_3 \sqsubseteq \forall r.(\neg B \sqcap \neg A)$

$\alpha_4 : A_4 \sqsubseteq \forall r.A \sqcap \forall s.B \sqcap A_5$

$\alpha_5 : A_5 \sqsubseteq \exists r.\neg A \sqcap A_6$

$\alpha_6 : A_6 \sqsubseteq \exists s.\neg B$

Consider the above example, by using standard DL TBox reasoning, it can be shown that the concept A_1 and A_4 are unsatisfiable. Analyzing concept A_1, the existing approaches identify $\{\alpha_1, \alpha_2, \alpha_3\}$ the only MUPS for A_1 in $\mathcal{T}_1$, but it is not clear whether A_2 or $\exists r.A_2$ of α_1 contradicts with α_3. In addition, it hides crucial information, e.g., that unsatisfiability of A_1 depends on all parts of α_2 or α_3. For A_4, $\{\alpha_4, \alpha_5\}$ is the only MUPS for A_4 in $\mathcal{T}_1$, which on behalf of the error caused by $\forall r.A$ of α_4 and $\exists r.\neg A$ of α_5. Actually, $\forall s.B$ of α_4, A_6 of α_5 and $\forall s.\neg B$ of α_6 also lead to the unsatisfiability of A_4, which would not be involved in the reason of unsatisfiability of A_4 since MUPS can not pinpoint the location of conflict, i.e., some unsatisfiable (or incoherent) reasons would be ignored by axiom pinpointing. We will use this example to explain our debugging methods.

3 Entailment and MUE

This section presents the main technical contribution of the paper. We would like to provide a framework for entailment-based axiom pinpointing. We will present the formal definitions which involve MUE, then show the relationship between MUPS and MUE. The second subsection is concerned with how to get all components w.r.t. axiom and terminology.

3.1 Formal Definitions

To compensate the limitations of axiom pinpointing, we introduce the notion of fine-grained axiom pinpointing and link it to description logic-based systems. Whereas the definitions of fine-grained axiom pinpointing are independent of the choice of a particular Description Logic.

Definition 2 (Entailment[17]). *Given a logical language $\mathcal{L}$, an entailment $\vDash$ states a relation between an terminology $\mathcal{T}$ and an axiom $\alpha \in \mathcal{L}$. We use $\mathcal{T} \vDash \alpha$ to denote that the ontology $\mathcal{T}$ entails the axiom α. Alternatively, we say that α is a consequence of the terminology $\mathcal{T}$ under entailment relation $\vDash$. The entailment relation is said to be a standard one if and only if α is always holds in any model in which the terminology $\mathcal{T}$ holds, i.e., for any model $\mathcal{I}$, $\mathcal{I} \vDash \mathcal{T} \Rightarrow \mathcal{I} \vDash \alpha$.*

Definition 3 (Atomic Entailment). *Let $\mathcal{T}$ be a terminology and β be an axiom such that $\mathcal{T} \vDash \beta$. We call $\vDash$ is an atomic entailment between $\mathcal{T}$ and β if $\{\beta\}$ has no consequence but β. Alternatively, β is an atomic consequence of $\mathcal{T}$.*

We denote by $\mathcal{E}(\mathcal{T})$ and $\mathcal{E}(\alpha)$ the set of all atomic consequences of terminology $\mathcal{T}$ and $\{\alpha\}$, respectively. If a terminology $\mathcal{T}$ is incoherent, then for any axiom

β, $\mathcal{T} \models \beta$, i.e., a standard entailment is explosive. Thus, we require $\mathcal{E}(\mathcal{T}) = \bigcup_{\alpha \in \mathcal{T}} \mathcal{E}(\alpha)$ if $\mathcal{T}$ is incoherent, and $\mathcal{E}(\alpha) = \{\alpha\}$ if the axiom α has no model. Intuitively, an atomic consequence of an axiom is a part of the axiom, and set of all atomic consequences of an axiom contains all parts of the axiom.

Definition 4 (MUE). *Let C be an unsatisfiable concept in terminology $\mathcal{T}$. A sub-TBox $\mathcal{T}_c \subseteq \mathcal{E}(\mathcal{T})$ is a minimal unsatisfiable entailment for C in $\mathcal{T}$ if C is unsatisfiable in $\mathcal{T}_c$, and C is satisfiable in every sub-TBox $\mathcal{T}'_c \subset \mathcal{T}_c$.*

The entailment-based axiom pinpointing inferential service is the problem of computing MUE. We denote by $mue(\mathcal{T}, C)$ the set of MUE of C in terminology $\mathcal{T}$. In the terminology of Reiter's diagnosis each $mue(\mathcal{T}, C)$ is a collection of conflict sets. The following are the MUE for our example TBox $\mathcal{T}_1$:

$$\begin{aligned} mue(\mathcal{T}_1, A_1) = \{&\{A_1 \sqsubseteq \exists r.A_2, A_1 \sqsubseteq A_3, A_2 \sqsubseteq A, A_3 \sqsubseteq \forall r.\neg A\},\\ &\{A_1 \sqsubseteq \exists r.A_2, A_1 \sqsubseteq A_3, A_2 \sqsubseteq B, A_3 \sqsubseteq \forall r.\neg B\}\}\\ mue(\mathcal{T}_1, A_4) = \{&\{A_4 \sqsubseteq \forall r.A, A_4 \sqsubseteq A_5, A_5 \sqsubseteq \exists r.\neg A\},\\ &\{A_4 \sqsubseteq \forall s.B, A_4 \sqsubseteq A_5, A_5 \sqsubseteq A_6, A_6 \sqsubseteq \exists s.\neg B\}\} \end{aligned}$$

MUE can be regarded as the fine-grained axiom pinpointing for MUPS. The relationship between axiom pinpointing and our pinpointing is established by Theorem 1, i.e., the one-to-many relationship between MUPS and MUE.

Theorem 1 (MUPS-to-MUEs relationship). *Let C be an unsatisfiable concept in terminology $\mathcal{T}$. Then:*
(1) If C is unsatisfiable in $\mathcal{T}$, then C is unsatisfiable in $\mathcal{E}(\mathcal{T})$.
(2) for every $\mathcal{M} \in mups(\mathcal{T}, C)$, there is a $\mathcal{K} \in mue(\mathcal{T}, C)$ s.t. $\mathcal{K} \subseteq \mathcal{E}(\mathcal{M})$.
(3) for any $\mathcal{M}_1, \mathcal{M}_2 \in mups(\mathcal{T}, C)$ and $\mathcal{K}_1, \mathcal{K}_2 \in mue(\mathcal{T}, C)$ where $\mathcal{M}_1 \neq \mathcal{M}_2$, $\mathcal{K}_1 \subseteq \mathcal{E}(\mathcal{M}_1)$ and $\mathcal{K}_2 \subseteq \mathcal{E}(\mathcal{M}_2)$, we have $\mathcal{K}_1 \neq \mathcal{K}_2$.

Proof. We prove (1), (2) and (3) in order.
(1) According to the definition of atomic entailment, If an axiom $\alpha \in \mathcal{T}$ has no model, $\mathcal{E}(\alpha)$=$\{\alpha\}$. Otherwise, we can prove $\{\alpha\}$ and $\mathcal{E}(\alpha)$ are equivalent with the axiom decomposition which is described in next subsection. In general, $\mathcal{T}$ and $\mathcal{E}(\mathcal{T})$ are equivalent.
(2) Since $\mathcal{M} \subseteq mups(\mathcal{T}, C)$, we have C is unsatisfiable in $\mathcal{E}(\mathcal{M})$. Thus, $mue(\mathcal{M}, C) = mups(\mathcal{E}(\mathcal{M}), C)$, and for every $\mathcal{K} \in mue(\mathcal{M}, C)$, we get $\mathcal{K} \subseteq \mathcal{E}(\mathcal{M})$.
(3) Suppose $\mathcal{K} \in mue(\mathcal{T}, C), \mathcal{M}_1, \mathcal{M}_2 \in mups(\mathcal{T}, C)$ $(\mathcal{M}_1 \neq \mathcal{M}_2)$ s.t. $\mathcal{K} \subseteq \mathcal{E}(\mathcal{M}_1)$ and $\mathcal{K} \subseteq \mathcal{E}(\mathcal{M}_2)$. Thus, there exists a sub-TBox $\mathcal{T}' \subseteq \mathcal{T}$ s.t. $\mathcal{K} \subseteq \mathcal{E}(\mathcal{T}')$ and $\mathcal{K} \nsubseteq \mathcal{E}(\mathcal{T}'')$ for every $\mathcal{T}'' \subset \mathcal{T}'$. Then C is unsatisfiable in $\mathcal{T}'$, $\mathcal{T}' \subseteq \mathcal{M}_1$ and $\mathcal{T}' \subseteq \mathcal{M}_2$. Since $\mathcal{M}_1, \mathcal{M}_2 \in mups(\mathcal{T}, C)$, we get $\mathcal{T}' = \mathcal{M}_1 = \mathcal{M}_2$ which contradicts with the assumption.

It is characteristic of our axiom pinpointing, in the sense to be made more precise, to uniquely identify each logical contradiction. For example, TBox $\mathcal{T} = \{A_1 \sqsubseteq A \sqcap B \sqcap A_2, A_2 \sqsubseteq \neg A \sqcap \neg B\}$, $\mathcal{T}$ is the only MUPS of A_1 while $mue(\mathcal{T}, A_1) = \{\{A_1 \sqsubseteq A, A_1 \sqsubseteq A_2, A_2 \sqsubseteq \neg A\}, \{A_1 \sqsubseteq B, A_1 \sqsubseteq A_2, A_2 \sqsubseteq \neg B\}\}$, which a MUPS has two MUE corresponding to. In this regard, entaiment-based axiom pinpointing is an extension of axiom pinpointing that MUE covers also the same unsatisfiable reasons of MUPS.

3.2 Syntactic Decomposition for Atomic Entailment

As previously mentioned, the theory of entailment-based axiom pinpointing is built on atomic entailment. For an incoherent terminology, we need to know the atomic entailments of each axiom instead. We give a syntactic decomposition notion to achieve this goal.

We give an overview of different kind of transformations that calculate the set of atomic entailment for an axiom in a terminology. Given a terminology $\mathcal{T}$ and an axiom $\alpha : C \sqsubseteq D$ where C is a atomic concept, apply the following transformation rules to α in each step (all rules of each step are correct[3]):

Step 1: (GCIs) Considering all such axioms $C_1 \sqsubseteq D_1, ..., C_n \sqsubseteq D_n$ in $\mathcal{T}$ where $C_i (1 \leq i \leq n)$ is a complex description, let $D' = (\neg C_1 \sqcup D_1) \sqcap ... \sqcap (\neg C_n \sqcup D_n)$, do $C' \sqsubseteq D \sqcap D'$, then transform D and D', respectively.

Step 2: (*Negation normal form*, NNF) Push all negation signs as far as possible into the description, using de Morgan's rules and usual rules for quantifiers[4].

Step 3: Repeated use of *distributive law*: $C_1 \sqcup (C_2 \sqcap C_3) = (C_1 \sqcup C_2) \sqcap (C_1 \sqcup C_3)$, $\forall R.(C_1 \sqcup (C_2 \sqcap C_3)) = \forall R.((C_1 \sqcup C_2) \sqcap (C_1 \sqcup C_3))$, $\exists R.(C_1 \sqcap (C_2 \sqcup C_3)) = \exists R.((C_1 \sqcap C_2) \sqcup (C_1 \sqcap C_3))$.

Step 4: Repeated use of following rules: $\forall R.(C_1 \sqcap C_2) = \forall R.C_1 \sqcap \forall R.C_2$, $\exists R.(C_1 \sqcup C_2) = \exists R.C_1 \sqcup \exists R.C_2$.

The transformation process always terminates and we end up with $D = D_1 \sqcap ... \sqcap D_m$ and $D' = D'_1 \sqcap ... \sqcap D'_n$ where constructor $\sqcap$ can only appear in $\lambda R.Y$ of $D_i (1 \leq i \leq m)$ and $D'_j (1 \leq j \leq n)$ while λ is constructor $\exists$, $\geq n$, or $\leq n$. Therefor, for any model $\mathcal{I}$, $\mathcal{I} \vDash \{\alpha : C \sqsubseteq D\} \Rightarrow \mathcal{I} \vDash C \sqsubseteq D_i (1 \leq i \leq m)$, and $C \sqsubseteq D_i$ has no entailment but itself. Consequently, $\{C \sqsubseteq D_1, ..., C \sqsubseteq D_m\}$ is the set of atomic entailment of α. Similarly, $\{C \sqsubseteq D_1, ..., C \sqsubseteq D_m, C \sqsubseteq D'_1, ..., C \sqsubseteq D'_n\}$ is the set of atomic entailment of α in terminology $\mathcal{T}$. Both the result of syntactic decomposition and the axiom have the same name and base symbols. Moreover, Since the result is obtained by a sequence of replacement steps, i.e., by replacing equals by equals. Therefore, $\mathcal{E}(\alpha)$ and α are syntactically and semantically equivalent, i.e., the result is the set of all atomic entailments of α. The atomic entailments of a terminology can be calculated by merging all axioms's. In example TBox $\mathcal{T}_1$, $\mathcal{E}(\alpha_1) = \{A_1 \sqsubseteq A_2,\ A_1 \sqsubseteq \exists r.A_2,\ A_1 \sqsubseteq A_3\}$.

On the other hand, using a rule $\mathcal{L} = \mathcal{R}$ above, it means $\mathcal{R}$ is obtained from $\mathcal{L}$. We can mark $\mathcal{R}$'s label is $\mathcal{L}$. Thus, keeping track of the transformations that occur during the processing step i.e. we can pinpoint the position of atomic entailment in original axiom.

[3] All these rules are correct and have been proved in the Description Logic Handbook[15].

[4] $\neg(\neg A) = A, \neg(\exists R.A) = \forall R.\neg A, \neg(\forall R.A) = \exists R.\neg A, \neg(\geq nR.A) = \leq (n-1)R.A, \neg(\leq nR.A) = \geq (n+1)R.A, \neg(C_1 \sqcup C_2) = \neg C_1 \sqcap \neg C_2, \neg(C_1 \sqcap C_2) = \neg C_1 \sqcup \neg C_2$.

4 Algorithms for Entailment-based Axiom Pinpointing

In this section, we discuss the algorithm for finding all MUE of an unsatisfiable concept. The algorithm we provide is reasoner-independent, in the sense that the DL reasoner is solely used as an oracle to determine concept satisfiability w.r.t a terminology. we provide the formal specification of the algorithm.

The ALL_MUE($\mathcal{T}, C, M, E$) algorithm receives a local terminology $\mathcal{T}$, a concept C, a local conflict M and a set of axioms E related to M directly[5], and outputs the set of all minimal subsets $\mathcal{T}' \subseteq \mathcal{E}(\mathcal{T} \cup E)$ such that C is unsatisfiable in $\mathcal{T}' \cup M$. The algorithm works in three main steps: first, it utilizes CONFLICT_HST to computes all related minimal contradiction of M from E for C; Then recursive call to the algorithm with the new parameters $\mathcal{T}$, M, and E for each related conflict we have obtained in previous step; Finally, combining the consequences of all recursive calls and obtain the final result. The loop in the second step is a main component of algorithm, which calculates the input parameters for next recursive call, it is mainly to do the following tasks: first of all, adding the obtained related conflict m to the original conflict M to get the new local conflict M'; Then, selecting axioms from $\mathcal{T}$ which is only related to the named symbols in m (because m has included all axioms related to M) dented by the new related axioms set A for the new local conflict M'; Last, get a new terminology $\mathcal{T}''$ by removing A from $\mathcal{T}$.

We can get all MUE of C in terminology $\mathcal{T}$ by calling ALL_MUE($\mathcal{T}, C, \varnothing, \varnothing$). Thus, the algorithm process guarantees three points as follows:

(a) Both axioms and named symbols of the input terminologies $\mathcal{T}$, M and E are mutually disjoint.
(b) $M \subseteq \mathcal{M}$ for every $\mathcal{M} \in mue(\mathcal{T} \cup M \cup E, C)$ if C is unsatisfiable in $\mathcal{T} \cup M \cup E$.
(c) C is satisfiable in $\mathcal{T} \cup M$ if M is not a MUE for C in $\mathcal{T} \cup M \cup E$.

Theorem 2. *Given an unsatisfiable concept C in a terminology $\mathcal{T}$, R returned by* ALL_MUE($\mathcal{T}, C, \varnothing, \varnothing$) *is the set of all* MUE *for C.*

Theorem 3. *The* CONFLICT_HST($\mathcal{T}, C, M, E$) *algorithm output all minimal subset $E' \subseteq E$ such that C is unsatisfiable in $\mathcal{T} \cup M \cup E'$, and C is satisfiable in $\mathcal{T} \cup M \cup E''$ for every $E'' \subset E'$.*

The CONFLICT_HST($N, F, HS, C, \mathcal{T}, M, E$) algorithm generates a Hitting Set Tree [16] with root node N, where a set F of conflict sets and a set HS of Hitting Sets are global, and outputs F. Initially, N, F and HS are empty, calling SINGLE_CONFLICT algorithm to get a value r for root node N if r is not empty. Then, generates the HST with root node N. In the loop, the algorithm generates a new node N' and a new edge e links N and N' in each iteration. Calling SINGLE_CONFLICT algorithm to obtain a value for the new node, and we mark the new node with $'\surd'$ if the value is empty.

[5] We say a TBox $\mathcal{T}'$ is directly related to the TBox $\mathcal{T}$ if all named symbols of $\mathcal{T}'$ is a subset of the signatures of $\mathcal{T}$.

Algorithm: ALL_MUE($\mathcal{T}, C, M, E$)
Input: a terminology $\mathcal{T}$, a concept C, a terminology M, a terminology E
Output: all minimal subsets of $\mathcal{E}(\mathcal{T} \cup E)$ conflict with M w.r.t. C R

$R \leftarrow \varnothing$;
$\mathcal{T}' \leftarrow \mathcal{T}$;
$E' \leftarrow E$;
if $M = \varnothing$ **then** /* The algorithm start with $M = \varnothing$ */
  $A \leftarrow \{\alpha \in \mathcal{T} \mid \alpha$ has the form $C \sqsubseteq D\}$;
  $\mathcal{T}' \leftarrow \mathcal{T} - A$;
  $E' \leftarrow \mathcal{E}(A)$;
if C is unsatisfiable in M **then** /* M is a MUE of C */
  $R \leftarrow \{M\}$;
  return R;
$H \leftarrow \varnothing$;
CONFLICT_HST($\varnothing, H, \varnothing, C, \mathcal{T}', M, E'$);
if $H = \varnothing$ **then** /* $H = \varnothing$ means M is a MUE of C */
  $R \leftarrow \{M\}$;
  return R;
for $m \in H$ **do**
  $M' \leftarrow M \cup m$; /* update the current MUE M of C */
  $S \leftarrow Sig(m) - \mathcal{D}(M') - \mathcal{B}(\mathcal{T}' \cup M')$;/*Select the related symbols of M'*/
  $A \leftarrow \{\alpha \in \mathcal{T}' \mid \alpha$ has the form $C' \sqsubseteq D'$ where $C' \in S\}$;
  $\mathcal{T}'' \leftarrow \mathcal{T}' - A$;
  $R' \leftarrow$ **ALL_MUE**($\mathcal{T}'', C, M', \mathcal{E}(A)$);
  $R \leftarrow R \cup R'$;
return R;

Two pruning strategy to the algorithm in order to reduce the size of HST and eliminate extraneous satisfiability tests. One is closing, if there exists a Hitting Set h in HS such that the path of N' is a superset of h, close node N' and the value is not computed for N' nor are any succor nodes generated, as indicted by a $'\times'$. The other one is reusing nodes: if there exists a node value k in F such that k and the path of N' are disjoint, set k as the value of N' directly without recalculation.

The SINGLE_CONFLICT($\mathcal{T}, C, M, E$) algorithm outputs a subset of E. In the loop, the algorithm removes an axiom from E in each iteration and check whether the concept C is satisfiable w.r.t. $\mathcal{T} \cup M \cup E$, in which case the axiom is added to R and reinserted into E. The process continues until all axioms in E have been tested. Finally, R is returned as output.

Theorem 4. *The* SINGLE_CONFLICT($\mathcal{T}, C, M, E$) *algorithm output a minimal subset $R \subseteq E$ such that C is unsatisfiable in $\mathcal{T} \cup M \cup R$, and C is satisfiable in $\mathcal{T} \cup M \cup R'$ for every $R' \subset R$.*

Proof. Let R be the output of algorithm SINGLE_CONFLICT($\mathcal{T}, C, M, E$). If C is satisfiable in $\mathcal{T} \cup M \cup E$, we get $R = \varnothing$. Otherwise, C is unsatisfiable in $\mathcal{T} \cup M \cup R$ upon termination. Suppose there exists a subset $R' \subset R$ such that C is unsatisfiable in $\mathcal{T} \cup M \cup R'$. Then, removing the axiom in $R - R'$ after the

Algorithm: CONFLICT_HST($N, F, HS, \mathcal{T}, C, M, E$)
Input: a node N, a set of conflict sets F, a set of hitting sets HS,
a terminology $\mathcal{T}$, a concept C, a terminology M, a terminology E
Output: input F

```
if N = ∅ then
    r ←SINGLE_CONFLICT(T, C, M, E);
    if r = ∅ then
        return;
    L(N) ← r;
    F ← {r};
for α ∈ L(N) do
    create a new node N′ and set L(N′) ← ∅;
    create a new edge e =< N, N′ > with L(e) ← α;
    if there exists a set h ∈ HS s.t. h ⊆ P(N) ∪ {α} then
        L(N′) ←′ ×′;
        continue;
    else if there exists a set k ∈ F s.t. k ∩ P(N) = ∅ then
            L(N′) ← k;
            CONFLICT_HST(N′, F, HS, T, C, M, E − {α});
    else
        m ← SINGLE_CONFLICT(T, C, M, E − {α});
        if m = ∅ then
            L(N′) ←′ √′;
            HS ← HS ∪ {P(N) ∪ {α}};
        else
            L(N′) ← m;
            F ← F ∪ {m};
            CONFLICT_HST(N′, F, HS, T, C, M, E − {α});
```

removal of R', we get C is satisfiable in $\mathcal{T} \cup M \cup R'$, which contradicts with the assumption.

Algorithm: SINGLE_CONFLICT($\mathcal{T}, C, M, E$)
Input: a terminology $\mathcal{T}$, a concept C, a terminology M, a terminology$E$
Output: a set of axioms (a subset of E) R

```
R ← ∅;
T′ ← T ∪ M ∪ E;
if (C is satisfiable in T′) then
    return R;
for α ∈ E do
    T′ ← T′ − {α};
    if C is satisfiable in T′ then
        T′ ← T′ ∪ {α};
        R ← R ∪ {α};
return R;
```

The problem of finding minimal Hitting Sets is known to be NP-COMPLETE, our algorithm is associated with the size of the element in $mue(\mathcal{T}, C)$. In this case, Let n be the cardinality of $mue(\mathcal{T}, C)$ and $S = \{k_1, ..., k_n\}$ be the value set of the size of its elements, the number of calls to SINGLE_CONFLICT and satisfiability tests involved is at most $k_1 \cdot ... \cdot k_n$.

5 Evaluation

Our algorithms for fine-grained axiom pinpointing have been realized in JAVA (JDK 1.6)using Pellet as the black-box reasoner. Tests are performed on a standard Windows operating system (Intel(R) Core(TM)i5-3470 CPU @ 3. 20GHz, 8. 00GB).

Before providing an evaluation of our algorithm, we briefly want to discuss a case study from Pizza. Then, we give the experimental results of common ontologies.

Example 2. $IceCream$ is an unsatisfiable concept in ontology $Pizza$.

$IceCream \sqsubseteq Food \sqcap \exists hasTopping.FruitTopping$	disjoint$(IceCream, Pizza)$
$FruitTopping \sqsubseteq PizzaTopping$	disjoint$(IceCream, PizzaTopping)$
$PizzaTopping \sqsubseteq Food$	role $hasTopping$: domain $Pizza$

$IceCream$ in $Pizza$ ontology has only one MUE $\mathcal{M}$:
$\mathcal{M} = \{IceCream \sqsubseteq \exists hasTopping.FruitTopping$, disjoint$(IceCream, Pizza)$, role $hasTopping$: domain $Pizza\}$

For unsatisfiable concept $IceCream$, taking away any single axiom from $\mathcal{M}$ makes $IceCream$ satisfiable, while $\mathcal{M}$ is not a MUPS since $IceCream \sqsubseteq \exists hasTopping.FruitTopping$ is only a part of original axiom of $IceCream$, which pinpoint the accurate component of contradiction within the axioms. As a consequence, the MUE indeed helped in some cases.

We have performed some preliminary experiments. We evaluated the method on five real-life OWL-DL ontologies vary in size, complexity and expressivity: Koala, MadCow, Pizza, MGED, DICE. The basic information of our experimental ontologies are depicted in Table 1, the results of test ontologies are summarized in Table 2.

According to Table 1 and Table 2, the results show that the scale of ontology and number of unsatisfiable concepts introduce an increase in the running time w.r.t. the fine-grained axiom pinpointing procedure. In the case of Koala and MadCow ontology, where the number of axioms related to an unsatisfiable concept are small (less than 10), the program ends in a very short period of time. However, for DICE ontology, where axioms responsible for an unsatisfiable concept are large in number (nearly 100), the running time of procedure is longer.

Table 1. The characteristics of test ontologies.

$\mathcal{T}$	$\mathcal{L}(\mathcal{T})$	$\lvert\mathcal{D}(\mathcal{T})\rvert$	$\lvert\mathcal{B}(\mathcal{T})\rvert$	$\lvert\mathbf{R}\rvert$	$\lvert\mathcal{T}\rvert$	$\lvert U_T\rvert$
Koala	ALCHON(D)	17	3	4	18	3
MadCow	ALCH(D)	40	13	16	41	1
Pizza	SHOIN	99	0	6	103	2
MGED	ALCH	231	2	110	231	32
DICE	ALCH	505	22	5	505	76

Note. Columns are: the terminology $\mathcal{T}$, the expressivity of termminology ($\mathcal{L}(\mathcal{T})$), number of named symbols ($|\mathcal{D}(\mathcal{T})|$), number of base symbols ($|\mathcal{B}(\mathcal{T})|$), number of roles ($|\mathbf{R}|$), number of axioms ($|\mathcal{T}|$), number of unsatisfiable concepts ($|U_T|$).

Table 2. The results of test ontologies.

$\mathcal{T}$	Koala	MadCow	Pizza	MGED	DICE
$\lvert$MUE($\mathcal{T}$)$\rvert$	3	1	2	57	79
Time(ms)	16	16	9	594	287177

Note. Rows are: the test terminology ($\mathcal{T}$), the sum of MUE of all unsatisfiable concepts in terminology ($|$MUE($\mathcal{T}$)$|$), the execution time of ALL_MUE algorithm for all unsatisfiable concepts in terminology where the unit is millisecond (Time).

6 Conclusion

In order to pinpoint debugging more accurately, we use the entailments to replace the corresponding axioms, then identify a minimal unsatisfiable subset of entailments for the new terminology. A new formal definition of MUE have be provided in this paper. At the same time, we presented a black-box pinpointing algorithm to solve it. Experimental results on common ontologies show that our axiom pinpointing provides incoherent terminology with more accurate incoherent reasons without losing contradictions masked by MUPS, and the performance of our algorithm is influenced by the size of related axioms directly. For future work, we plan to adopt the dependency between concepts, investigate different kinds of selection function, that hopefully improve the efficiency of entailment-based axiom pinpointing.

Acknowledgments. This work was supported in part by NSFC under Grant Nos. 61272208, 61133011, 41172294, 61170092; Jilin Province Science and Technology Development Plan under Grant Nos.201201011.

References

1. Lambrix, P., Qi, G.L., Horridge. M., (eds): Proceedings of the First International Workshop on Debugging Ontologies and Ontology Mappings, (Galway, Ireland), LECP 79, Linköping University Electronic Press (2012)
2. Baader, F., Rafael Peñaloza, R.: Automata-Based Axiom Pinpointing. In: 4th international joint conference on Automated Reasoning, pp. 226-241. Springer-Verlag Berlin, Heidelberg (2008)
3. Meyer,T., Lee, K., Booth,R., Pan, J.Z.: Finding Maximally Satisfiable Terminologies for the Description Logic ALC. In: 21st National Conference on Artificial Intelligence, pp. 269-274. AAAI press (2006)
4. Lam, S.C., Pan, J.Z., Sleeman, D., Vasconcelos, W.: A Fine-Grained Approach to Resolving Unsatisfiable Ontologies. In: 2006 IEEE/WIC/ACM International Conference on Web Intelligence, pp. 428-434. IEEE Computer Society Washington (2006)
5. Schlobach, S., Cornet, R.: Non-Standard Reasoning Services for the Debugging of Description Logic Terminologies. In: 18th international joint conference on Artificial intelligence, pp. 355-360. Morgan Kaufmann Publishers Inc (2003)
6. Schlobach, S.: Diagnosing Terminologies. In: 20th national conference on Artificial intelligence, pp. 670-675. AAAI Press (2005)
7. Schlobach, S., Huang, Z.S., Cornet, R., Harmelen, F.V.: Debugging Incoherent Terminologies. J. Automated Reasoning. 39(3), 317-349 (2007)
8. Parsia, B., Sirin, E., Kalyanpur, A.: Debugging OWL Ontologies. In: 14th International Conference on World Wide Web, pp. 633-640. ACM Press, New York (2005)
9. Kalyanpur, A., Parsia, B., Sirin, E., Hendler, J.: Debugging Unsatisfiable Concepts in OWL Ontologies. J. Web Semantics. 3(4), 268-293 (2005)
10. Kalyanpur, A., Parsia, B., Horridge, M., Sirin, E.: Finding All Justifications of OWL DL Entailments. In: 6th international The semantic web and 2nd Asian conference on Asian semantic web conference, pp. 267-280. Springer-Verlag Berlin, Heidelberg (2007)
11. Wang, H., Horridge, M., Rector, A., Drummond, N., Seidenberg. J.: Debugging OWL-DL Ontologies: A Heuristic Approach. In: 4th International Semantic Web Conference, pp. 745-757. Springer-Verlag (2005)
12. Du, J.F., Qi, G.L., Ji, Q.: Goal-Directed Module Extraction for Explaining OWL DL Entailments. In: 8th International Semantic Web Conference, pp. 163-179. Springer-Verlag (2009)
13. Haase, P., Qi, G.L.: An Analysis of Approaches to Resolving Inconsistencies in DL-based Ontologies. In: International Workshop on Ontology Dynamics, pp. 97-109 (2007)
14. Horridge, M., Parsia, B., Sattler, U.: Justification Masking in Ontologies. In: 13th International Conference on the Principles of Knowledge Representation and Reasoning (2012)
15. Baader, F., Calvanese, D., McGuinness, D., et al.: The Description Logic Handbook: Theory, Implementation and Application. Second Edition. Cambridge University Press (2007)
16. Reiter, R.: A Theory of Diagnosis from First Principles. J. Artificial Intelligence, 32(1), 57-95 (1987)
17. Haase, P., Harmelen, F.V., Huang, Z.S., Stuckenschmidt, H., Sure, Y.: A Framework for Handling Inconsistency in Changing Ontologies. In: 4th International Semantic Web Conference, pp. 353-367. Springer-Verlag Berlin, Heidelberg (2005)

An automatic way of generating incoherent terminologies with parameters

Yu Zhang, Dantong Ouyang, and Yuxin Ye *

College of Computer Science and Technology,
Jilin University, Changchun 130012, China

Abstract. The minimal incoherence preserving sub-terminologies (Mips) is defined for identifying the axioms responsible for the unsatisfiable concepts in incoherent ontology. While a great many performance evaluations have been proposed in the past, what remains to be investigated is whether we have effective reasoners to solve the Mips problems, in which case a particular reasoner will be more efficiency than others. After analyzing the structural complexity of terminology, we develop a Mips Benchmark (MipsBM) to evaluate the performances of reasoners by defining six complexity metrics based on concept dependency networks model. Evaluation experiments show that the proposed metrics can effectively reflect the complexity of benchmark data. Not only can the benchmark help the users to determine which reasoner is likely to perform best in their applications, but also help the developers to improve the performances and qualities of their reasoners.

Keywords: Incoherent terminology; Mips; Benchmark; MipsBM

1 Introduction

In practice, building an ontology is a very complicated process and is easy to make errors, an ontology $\mathcal{O}$ is incoherent if there exists an unsatisfiable concept in $\mathcal{O}$, and the existence of unsatisfiable concept indicates that the formal definition is incorrect. Therefore, how to find all the unsatisfiable concepts is the challenging of ontology debugging. Researchers have proposed various methods to debug incoherent ontology. Ontology debugging is achieved by using reasoners, currently, most of the reasoners, such as Pellet [1], HermiT[2], FaCT++[3], TrOWL[4] and JFact support the inference tasks. A great many performance evaluations for reasoners have been performed in the past, What remains to be investigated is whether we have effective reasoners to solve the Mips problems, in which cases a particular reasoner will be more efficiency than others. There are several criteria for a good benchmark test data. First, we need to systematically construct several types of logical contradictions to create an incoherent TBox. Second, there must be a number of parameters that could influence the complexity of benchmark data and the difficulty for reasoning.

* corresponding author: yeyx@jlu.edu.cn (Yuxin Ye).

2 Related Work

In the research of knowledge base query, (LUBM) [5] is developed based on several complexity metrics of ontology and provides 14 test queries to assess the efficiency, correctness and completeness of the knowledge base. However, the correlations between the classes of LUBM are low, thus Li Ma extends it to University Ontology Benchmark (UOBM) [6] by adding a series of association classes. However, either LUBM or UOBM only can evaluate single ontology, thus Yingjie Li et al. [7] develop a multi-ontology synthetic benchmark that can evaluate not only single ontology but also federated ontologies. In the research of ontology matching, Alfio Ferrara et al. [8] propose a disciplined approach to the semi-automatic generation of benchmarks called SWING (Semantic Web Instance Generation), but all the evaluations in SWING are only for single language, so Christian Meilicke et al. [9] design a benchmark for multilingual ontology matching called MultiFarm. Besides, the work in [10] presents the design of a modular test generator to evaluate different matchers on the generated tests. In the research of ontology reasoning and debugging, the benchmarks proposed in [11] and [12] are used to evaluate the classification performances of reasoners. The work in [13] focus on the applicability of specific reasoners to certain expressivity clusters, and evaluate the loading time, classification and conjunctive queries performances of reasoners. JustBench [14] is a typical benchmark to evaluate the reasoners for calcuating justification. In [15], several machine learning techniques are used to predict classification time and determine the metrics that can be used to predict reasoning performance. The work in [16] proposes a method to construct the justification dataset from realistic ontologies with different sizes and expressivities.

3 Complexity Analysis for Incoherent TBox

The expressivity of a particular DL is determined by the concept constructors it provides [17]. $\mathcal{SHOIN}(\mathcal{D})$ is a very expressive DL that provides the constructors including $\mathcal{H}$ (role hierarchies), $\mathcal{O}$(nominals), $\mathcal{I}$(inverse roles), $\mathcal{N}$(Number restriction) and $\mathcal{S}$ is the abbreviation for $\mathcal{ALC}$ with transitive roles. $\mathcal{ALC}$ is the basic description logic consisting of the constructors $\neg C$ (negation), $C \sqcap D$(conjunction), $C \sqcup D$(disjunction), $\exists r.C$(existential restriction) and $\forall r.C$(value restriction).

Example 1. Suppose a $\mathcal{SHOIN}(\mathcal{D})$ TBox contains the following axioms:
$\alpha_1 : S_1 \equiv \exists r_1.B_1$, $\alpha_2 : S_2 \equiv \forall r_2.B_2$, $\alpha_3 : S_3 \equiv \{B_2\} \sqcup \{B_3\}$,
$\alpha_4 : S_4 \sqsubseteq S_1 \sqcup S_2$, $\alpha_5 : S_5 \sqsubseteq \exists r_4.S_3 \sqcap S_4$, $\alpha_6 : S_6 \equiv \exists r_5.B_4$,
$\alpha_7 : S_7 \equiv \{B_4\} \sqcup \{B_5\}$, $\alpha_8 : S_8 \sqsubseteq \exists r_6.S_6 \sqcap S_7$, $\alpha_9 : C_1 \equiv \forall t_1.\neg A_1$,
$\alpha_{10} : C_2 \equiv \exists t_1.A_1 \sqcap \forall t_2.A_2$, $\alpha_{11} : C_3 \equiv \forall t_2.A_2 \sqcap \exists t_2.\neg A_2$, $\alpha_{12} : C_4 \equiv \exists t_2.\neg A_2$,
$\alpha_{13} : C_5 \equiv \geq 3t_3 \sqcap \leq 2t_3$, $\alpha_{14} : C_6 \sqsubseteq C_1 \sqcap C_2$, $\alpha_{15} : C_7 \sqsubseteq C_2$,
$\alpha_{16} : C_8 \sqsubseteq C_4$, $\alpha_{17} : C_9 \sqsubseteq C_7 \sqcap C_8$, $\alpha_{18} : C_{10} \sqsubseteq C_5 \sqcap C_6 \sqcap C_9$,
$\alpha_{19} : r_1 \equiv r_4^-$, $\alpha_{20} : r_2 \sqsubseteq r_3$, $\alpha_{21} : r_5 \circ r_5 \sqsubseteq r_5$

Stefan Schlobach proposes the minimal unsatisfiability preserving sub-TBox (Mups)[18] to identify the axioms responsible for the unsatisfiability of concepts in incoherent TBox. For $\mathcal{T}$ in example 1, it can be shown that the concepts $C_3, C_5, C_6, C_9, C_{10}$ are unsatisfiable by using standard DL TBox reasoning. We can get their Mups:

$\text{Mups}(\mathcal{T}, C_3) = \{\{\alpha_{11}\}\}$, $\quad \text{Mups}(\mathcal{T}, C_5) = \{\{\alpha_{13}\}\}$,
$\text{Mups}(\mathcal{T}, C_6) = \{\{\alpha_9, \alpha_{10}, \alpha_{14}\}\}$, $\quad \text{Mups}(\mathcal{T}, C_9) = \{\{\alpha_{10}, \alpha_{12}, \alpha_{15}, \alpha_{16}, \alpha_{17}\}\}$,
$\text{Mups}(\mathcal{T}, C_{10}) = \{\{\alpha_{13}, \alpha_{18}\}, \{\alpha_9, \alpha_{10}, \alpha_{14}, \alpha_{18}\}, \{\alpha_{10}, \alpha_{12}, \alpha_{15}, \alpha_{16}, \alpha_{17}, \alpha_{18}\}\}$.

Definition 1 (MIPS[18]). *A TBox $\mathcal{T}' \subseteq \mathcal{T}$ is a minimal incoherence preserving sub-TBox* (MIPS) *of $\mathcal{T}$ if and only if $\mathcal{T}'$ is incoherent, and every sub-TBox $\mathcal{T}'' \subset \mathcal{T}'$ is coherent. The set of all* MIPS *of $\mathcal{T}$ is denoted as MIPS($\mathcal{T}$).*

We will abbreviate the set of MIPS for $\mathcal{T}$ by Mips($\mathcal{T}$). For $\mathcal{T}$ in example 1 we can get $\text{Mips}(\mathcal{T}) = \{\{\alpha_{11}\}, \{\alpha_{13}\}, \{\alpha_9, \alpha_{10}, \alpha_{14}\}, \{\alpha_{10}, \alpha_{12}, \alpha_{15}, \alpha_{16}, \alpha_{17}\}\}$.

Definition 2 (Mips Size). *Let Mips($\mathcal{T}$) be the Mips of an incoherent TBox $\mathcal{T}$, the number of axiom set in the Mips($\mathcal{T}$) is called Mips Size.*

Let *Ms* represent the Mips size, for $\text{Mips}(\mathcal{T}) = \{\{\alpha_{11}\}, \{\alpha_{13}\}, \{\alpha_9, \alpha_{10}, \alpha_{14}\}, \{\alpha_{10}, \alpha_{12}, \alpha_{15}, \alpha_{16}, \alpha_{17}\}\}$, there are four axiom sets in the Mips($\mathcal{T}$), thus the Mips size *Ms* =4.

Definition 3 (Mips Depth). *Let Mips($\mathcal{T}$) be the Mips of an incoherent TBox $\mathcal{T}$, the maximum number of axioms in all the axiom sets is called Mips Depth.*

Let *Md* represent the Mips depth. Using the previous example again, both the number of axioms in the first axiom set $\{\alpha_{11}\}$ and the second axiom set $\{\alpha_{13}\}$ are one, while in the third axiom set $\{\alpha_9, \alpha_{10}, \alpha_{14}\}$, the number is three, and in the last axiom set $\{\alpha_{10}, \alpha_{12}, \alpha_{15}, \alpha_{16}, \alpha_{17}\}\}$, the number is five, thus the maximum number of axioms *Md*=5.

Given a TBox $\mathcal{T}$, the concept dependency networks N are defined as follows.

Definition 4 (concept dependency networks). *A directed graph N=(V,E) is a corresponding concept dependency networks of a given TBox $\mathcal{T}$, where V is the set of vertices representing all the concepts in $\mathcal{T}$, and E is the set of edges representing all the dependencies between concepts.*

Figure 1 represents the concept dependency networks of TBox $\mathcal{T}$ in Example 1. On the basis of the concept dependency networks model, the semantic dependency of concept can be defined as follows.

Definition 5 (concept depth). *In the concept dependency networks of TBox $\mathcal{T}$, suppose the concept depth of C is cd(C), cd(C) can be recursively defined as follows.*
if $C \doteq C_1 \sqcap C_2$, then $dep(C) = max(cd(C_1), cd(C_2)) + 1$;
if $C \doteq C_1 \sqcup C_2$, then $dep(C) = max(cd(C_1), cd(C_2)) + 1$;

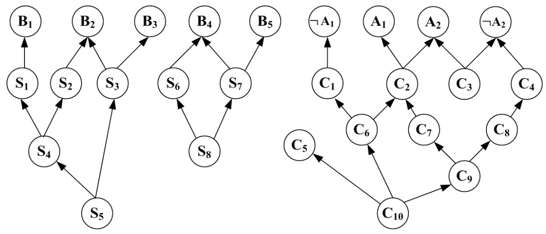


Fig. 1. concept dependency networks of TBox $\mathcal{T}$.

if $C \doteq \exists r.C_1$*, then* $cd(C) = cd(C_1) + 1$*;*
if $C \doteq \forall r.C_1$*, then* $cd(C) = cd(C_1) + 1$*;*
if $C \doteq C_1$*, then* $cd(C) = cd(C_1) + 1$*;*
if $C \doteq \neg C_1$*, then* $dep(C) = cd(C_1) + 1$*;*
if C *is an atom, then* $cd(C) = 0$*;*
The $\doteq$ *is either* $\equiv$ *or* $\sqsubseteq$*.*

If the concept depth of C is 1, C is called a simple concept, otherwise called a complex concept. Suppose that TBox $\mathcal{T}$ contains p simple concepts and q complex concepts, we have the total number of concepts $m = p+q$. Besides, the maximal concept depth of $\mathcal{T}$, denoted as λ, can be defined as: $\lambda = max(cd(C_i)), 1 \leq i \leq m$.

Definition 6 (semantic cluster). *In the TBox* $\mathcal{T}$*, the subTBox* $\mathcal{T}' \subseteq \mathcal{T}$ *which is composed of concepts linked together by semantic dependency relationship, is called a semantic cluster of* $\mathcal{T}$*.*

Suppose that the number of semantic dependency is μ. The semantic cluster must satisfy the constraint $p+\mu\sum_{i=1}^{\lambda} dep(C_i) = m$. Furthermore, the clustering coefficient can be defined as:

$$\eta = \frac{\mu \sum_{i=1}^{\lambda} dep(C_i)}{m}. \tag{1}$$

If $\mu = 0$, which means there is not any semantic cluster in the TBox, so the minimum of clustering coefficient $\eta_{min} = 0$. If, however, $p = 0$, which means the TBox is composed only of complex concepts, then $\mu\sum_{i=1}^{\lambda} dep(C_i) = m$, so the maximum of clustering coefficient $\eta_{max} = 1$.

4 MipsBM System

MipsBM consists of two components: satisfiable concept generator and unsatisfiable concept generator. According to the characteristics of axioms appearing in $\mathcal{SHOIN}(\mathcal{D})$ TBox, we categorize them into two groups: constructors and operands. The constructors group consists of concept constructor and property constructor. And the operands group is composed of atom set and role set. The constructors and operands table are shown in Table 1.

Table 1. Constructors and Operands Table

Satisfiable Constructors		Unsatisfiable Constructors	
SatConceptConstructor	Syntax	unSatConceptConstructor	Syntax
subClass	$S_1 \sqsubseteq S_2$	subClass	$C_1 \sqsubseteq C_2$
equivalentClass	$S_1 \equiv S_2$	equivalentClass	$C_1 \equiv C_2$
intersection	$S_1 \sqcap S_2$	intersection	$C1 \sqcap C_2$
allValues	$\forall r_1.B_1$	allValues	$\forall t_1.A_1$
someValues	$\exists r_2.B_2$	someValues	$\exists t_2.A_2$
union	$S_1 \sqcup S_2$	complement	$\neg A$
oneOf	$\{x_1, x_2, x_3\}$	disjointWith	$C_1 \sqsubseteq \neg C_2$
PropertyConstructor	Syntax	maxCardinality	$\geq nr_1$
subProperty	$r_1 \sqsubseteq r_2$	minCardinality	$\leq nr_1$
equivalentProperty	$r_1 \equiv r_2$	PropertyConstructor	Syntax
TransitiveProperty	$r_1^+ \sqsubseteq r_2$	subProperty	$t_1 \sqsubseteq t_2$
inverseOf	r^-	equivalentProperty	$t_1 \equiv t_2$
Satisfiable Operands		Unsatisfiable Operands	
SatAtomSet	$B_1, B_2, \cdots, B_m$	unSatAtomSet	$A_1, A_2, \cdots, A_n$
SatRoleSet	$r_1, r_2, \cdots, r_m$	unSatRoleSet	$t_1, t_2, \cdots, t_n$

Algorithm 1: satGenerator

Input: *satnum*: number of satisfiable concepts
μ: number of semantic clusters
λ: maximum concept depth
output: S: satisfiable concept set

while$(\mu > 0)$
  $constructor_k = randomSelect(SatConceptConstructor)$;
  $op_k = randomSelect(SatAtomSet, SatRoleSet)$;
  $S_k = generateAxiom(constructor_k, op_k), S.add(S_k), k++$;
  while$(\lambda > 0)$
    $conceptstructor_k = randomSelect(SatConceptConstructor)$;
    $propstructor_k = randomSelect(SatPropertyConstructor)$;
    $op_k = randomSelect(SatAtomSet, SatRoleSet), op_k = op_k \cup S_k$;
    $\{S_k, P_k\} = generateAxioms(conceptstructor_k, propstructor_k, op_k)$;
    $S = S \cup \{S_k\}, \lambda --, k++$;
  $\mu --$;
$num = satnum - size(S)$;
while$(num \geq 0)$
  $constructor_k = randomSelect(SatConceptConstructor)$;
  $op_k = randomSelect(SatAtomSet, SatRoleSet)$;
  $S_k = generateAxiom(constructor_k, op_k)$;
return S

The proof for Algorithm 1 is as follows.

Proof. Because there are not any complement or disjoint constructors in the Satisfiable Constructors in Table 1, the concepts generated by Algorithm 1 must be satisfiable.

The first while loop corresponds to the number of semantic clusters, in each loop, the algorithm creates a semantic cluster, and the value of μ is decreased by 1 until $\mu = 0$. The second while loop corresponds to the maximum concept depth, in each loop, the algorithm creates a concept, and the concept depth of the latter concept is 1 bigger than that of the former one. When the loop is finished, the concept depth of the last concept reaches λ. After that, the number of satisfiable concepts is obtained, the rest of the concepts are created in the third while loop.

In order to build an incoherent terminology, MipsBM needs to create several unsatisfiable concepts which can be achieved through systematically constructing logical clashes.

Definition 7 (Independent Unsatisfiable Concept). *C is an independent unsatisfiable concept if the unsatisfiability of C depends on the concept definition rather than the unsatisfiability of other concepts.*

Definition 8 (Dependent Unsatisfiable Concept). *C is a dependent unsatisfiable concept if the unsatisfiability of C depends on the unsatisfiability of other concepts.*

From the Example 1, C_3, C_5, C_6 and C_9 are independent unsatisfiable concepts, C_{10} is dependent unsatisfiable concept because its unsatisfiability depends on unsatisfiable concepts C_5, C_6, C_9.

Definition 9 (Clash Sequences). *Let Seq$^+$(C) be the positive clash sequence of C, and Seq$^-$(C) the negative clash sequence. Seq$^+$(C) is of the form $< (C_1, I_1, C_2), (C_2, I_2, C_3), \cdots, (C_m, I_m, C) >$ $(i = 1, \cdots, m)$, where $C_i \sqsubseteq C_{i-1}$, I_i represents the indexes of axioms related to $C_i \sqsubseteq C_{i-1}$. Seq$^-$(C) is of the form $< (\neg C_1, I'_1, C'_2), (C'_2, I'_2, C'_3), \cdots, (C'_n, I'_n, C) >$ $(i = 1, \cdots, n)$, where $C'_i \sqsubseteq C'_{i-1}$, I'_i represents the indexes of axioms related to $C'_i \sqsubseteq C'_{i-1}$. After that, the unsatisfiable concept C can be generated by $C \sqsubseteq C_m \sqcap C'_n$.*

For example, The clash sequences of C_9:
Seq$^+(C_9)$=$< (A_1, \{\alpha_{10}\}, C_2), (C_2, \{\alpha_{10}, \alpha_{15}\}, C_7), (C_7, \{\alpha_{10}, \alpha_{15}, \alpha_{17}\}, C_9)) >$,
Seq$^-(C_9)$=$< (\neg A_1, \{\alpha_{12}\}, C_4), (C_4, \{\alpha_{12}, \alpha_{16}\}, C_8), (C_8, \{\alpha_{12}, \alpha_{16}, \alpha_{17}\}, C_9) >$.
Unsatisfiable concepts can be divided into two types as follows.

complement clash: C is a *complement clash* concept if it is a subclass of both class A and the complement of class A. For example:
$\alpha_1 : C_1 \sqsubseteq \forall t_1.A_1 \sqcap \exists t_1.\neg A_1$. Then C_1 is a *complement clash* root concept.

cardinality clash: C is a *cardinality clash* concept if the at-least restriction is bigger than the at-most restriction in its definition. For example:
$\alpha_2 : C_2 \equiv \geq 2.t_2 \sqcap \leq 1.t_2$. Then C_2 is a *cardinality clash* root concept.

Unsatisfiable concept generator (Algorithm 2) creates the satisfiable concepts by constructing clash sequences.

Algorithm 2: unsatGenerator(*unsatnum*,*Ms*,*iMd*)

inputs:*unsatnum*: number of unsatisfiable concept
Ms: Mips size; *iMd*: increasement of Mips depth
output: U: unsatisfiable concept set; Mips($\mathcal{T}$): the Mips of $\mathcal{T}$

$U = \varnothing$, Mips($\mathcal{T}$) $= \varnothing, k = 0, len = 0$;
constructor = *randomSelect(UnsatConceptConstructor*;)
construct a pair of clsh sequences : $\{Seq^+, Seq^+\}$
$D_0 \leftarrow Seq^+$, $D'_0 \leftarrow Seq^-$;
$I_{(C_k)} : C_k \doteq intersectionOf(D_0, D'_0)$;
$C_R.add(C_k)$, $Mips.add(I_{(C_k)})$, k++, *len*++;
while($k \leq Ms$)
    len=*len*+*iMd*;
    construct a pair of clsh sequences : $\{Seq^+, Seq^+\}$
    $D_0 \leftarrow Seq^+$, $D'_0 \leftarrow Seq^-$;
    for(i=j=1; $j < len$; i++,j=j+2)
        $S_{x,y} \leftarrow$ *(SatAtomSet,SatRoleSet,someValues,allValues)*;
        $I_{(D_i)} : D_i \doteq intersectionOf(D_{i-1}, S_x)$;
        $I'_{(D'_i)} : D'_i \doteq intersectionOf(D'_{i-1}, S_y)$;
        $Mips.add(I_{(D_i)}, I'_{(D'_i)})$;
    $I_{(C_k)} : C_k \doteq intersectionOf(D_i, D'_i)$, $C_R.add(C_k)$, $Mips.add(I_{(C_k)})$;
    U.add(C_R), Mips($\mathcal{T}$).add(*Mips*), k++;
$num = unsatnum - k$;
while($m \leq num$)
    $C_r \leftarrow randomSelect(C_R)$;
    $S_z \leftarrow$ *(SatAtomSet,SatRoleSet,someValues,allValues)*;
    $C_k \doteq intersectionOf(C_r, S_z)$;
    U.add(C_k),$m++$;
return $U, Mips(\mathcal{T})$

Theorem 1 *The unions of clash sequences of independent unsatisfiable concepts are the Mips of TBox.*

Proof. By Definition 1(Incoherent TBox), we have that a TBox $\mathcal{T}$ is incoherent if and only if there is a concept name in $\mathcal{T}$ which is unsatisfiable. Therefore, according to Definition 3(Mips), we can prove Theorem 1 based on two points:

■ One concept is unsatisfiable in the union of contradiction sequences.

■ And the concept is satisfiable in every subset of the union of contradiction sequences.

We prove the first point. Let C_k be a satisfiable concept, According to the unsatGenerator algorithm, C_k is created by $C_k \sqsubseteq D_i \sqcap D'_i$, where $D_i \sqsubseteq D_{i-1}$ and $D'_i \sqsubseteq D'_{i-1}$. Similarly, $D_{i-1} \sqsubseteq D_{i-2}, \cdots, D_2 \sqsubseteq D_1$ and $D'_{i-1} \sqsubseteq D'_{i-2}, \cdots, D'_2 \sqsubseteq D'_1$. The corresponding clash sequences are:

$<(D_1, I_1, D_2), (D_2, I_2, D_3), \cdots, (D_i, I_i, C_k)>$, where $I_i = I_i \cup I_{i-1}$.

$<(D'_1, I'_1, D'_2), (D'_2, I'_2, D'_3), \cdots, (D'_i, I'_i, C_k)>$, where $I'_i = I'_i \cup I'_{i-1}$.

D_1 and D'_1 have the form either $D_1 \equiv A, D'_1 \equiv \neg A$ or $D_1 \equiv \geq mt, D'_1 \equiv \leq nt (m > n$, and t is a role name). this implies that $C_k \sqsubseteq D_1$ and $C_k \sqsubseteq D'_1$, i.e.

$C_k \sqsubseteq A \sqcap \neg A$ or $C_k \sqsubseteq \geq mt \sqcap \leq nt(m > n)$. Therefore, C_k is unsatisfiable in $\mathcal{T}' = I_i \cup I_i'$, i.e. C_k is unsatisfiable in the union of clash sequences.

Next, we prove the second point. Let $\mathcal{T}''$ be the every subset of $\mathcal{T}'$ after removing any one axiom α_j from $I_i \cup I_i'$. If α_j occurs in the Seq^+ of C_k, we have that $D_i \sqcap D_{i-1}, D_{i-1} \sqsubseteq D_{i-2}, \cdots, \alpha_j : D_j \sqsubseteq D_{j-1}, \cdots, D_2 \sqsubseteq D_1$. Removing α_j is equivalent to removing $D_j \sqsubseteq D_{j-1}$ from the Seq$^+$ of C_k, so D_i is not the subset of D_1. If α_j occurs in the Seq^- of C_k, we have that $D_i' \sqcap D_{i-1}', D_{i-1}' \sqsubseteq D_{i-2}', \cdots, \alpha_j : D_j' \sqsubseteq D_{j-1}', \cdots, D_2' \sqsubseteq D_1'$. Removing α_j is equivalent to removing $D_j' \sqsubseteq D_{j-1}'$ from the Seq$^-$ of C_k, so D_i' is not the subset of D_1'. We know $C_k \sqsubseteq D_i \sqcap D_i'$, so C_k is not the subset of both D_1 and D_1'. Therefore, C_k is satisfiable in $\mathcal{T}''$, i.e. C_k is satisfiable in every subset of the union of clash sequences.

5 Evaluation with MipsBM

The MipsBM experiments demonstrate how to evaluate the performances of reasoners for Calculating Mips. Pellet 2.3.1 [1], HermiT 1.3.8 [2], FaCT++ 1.6.2 [3], JFact 1.0.0 [4] and TrOWL 1.4 [5] are the five most widely-used description logics reasoners used in our experiments. The tests are performed on a PC (Intel(R) Core(TM) CPU 3.40Ghz) with 4 GB RAM. Our performance measure is the run time (in seconds) to calculate Mips.

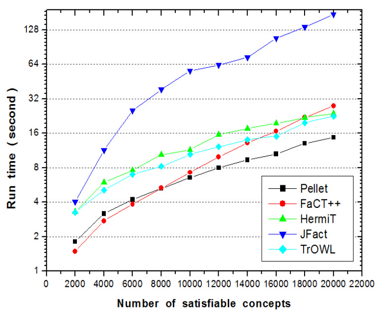


Fig. 2. evaluations for the number of satisfiable concepts

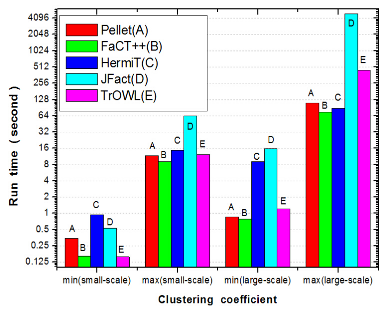
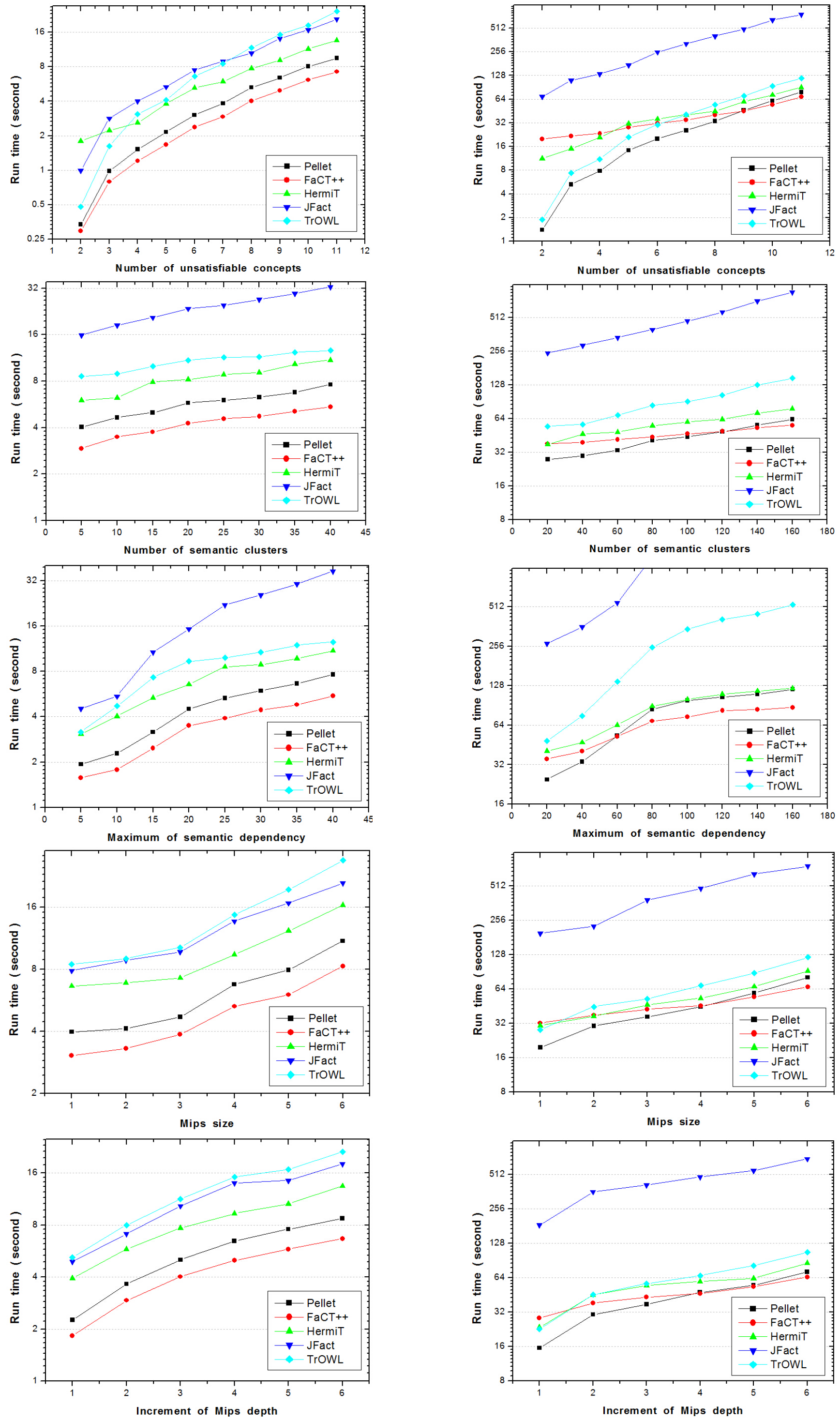


Fig. 3. evaluations for the clustering coefficient

From Figure 2, we can conclude from the second experiment that TBox size plays a significant influence on the performances of different reasoners. Therefore, the following evaluations can be viewed from two aspects: small scale TBox (the number of concepts $m = 2000$) and large scale TBox ($m = 20000$).

[1] http://clarkparsia.com/pellet
[2] http://www.hermit-reasoner.com/
[3] http://code.google.com/p/factplusplus/
[4] http://sourceforge.net/projects/jfact/
[5] http://trowl.eu/



In the case of large scale TBox　　In the case of small scale TBox

Fig. 4. performance evaluations for reasoners about complexity metrics

After the evaluation experiments, we give a further analysis from two perspectives.

What makes an incoherent TBox difficult to calculate Mips? In order to answer this question, we consider the impact of construction parameters on structure complexity of incoherent terminology. A large number of satisfiable concepts mean a large size of TBox, Reasoners have to take a lot of time to perform satisfiability checking, so the run time becomes longer. There are many relevance relations between one concept and others if the concept depth is large, as the number of semantic clusters increases, the number of semantic dependencies between the concepts will grow significantly. The Mips size corresponds to the scale of minimal conflict axiom set, our reasoners need to find the minimal conflict axiom set of the incoherent TBox, thus the size of semantic dependency is strictly determined by the Mips depth. According to Definition 9, the clash sequences of unsatisfiable concepts correspond to the increase of Mips depth, the larger the depth is, the longer the clash sequences are, therefore, a larger value of the increase of Mips depth leads to a higher complex of incoherent TBox.

Which is the most appropriate reasoner to solve Mips problem? Because of the differences of optimization approaches, the five reasoners have different performances for the same benchmark test data. When the number reaches 8000, Pellet is faster than FaCT++, when reaches 14000, TrOWL is faster than FaCT++, and when reaches 18000, HermiT performs better than FaCT++. In the process of consistency checking, HermiT uses the anywhere blocking technique to limit the sizes of models which are constructed, so it has an advantage over ABox. Unfortunately, the ontology test data generated by our MipsBM only consists of TBox, thus the advantages haven't been fully fulfilled. Our experiments show that timeout is the main reason to cause the failures of JFact, especially for a large inputs, It is because JFact takes longer to load the TBox than others. In the case of large scale TBox, JFact fails to resolve the Mips problems when the number of clusters increases beyond 80 in the fourth experiment.

6 Conclusion and future work

This paper presents a benchmark to generate different complicated terminologies to evaluate the performances of description logics reasoners for calculating Mips. Our purpose is to find out the reasons which result in the difficulty and high cost of ontology debugging. Experiments show that the six construction parameters can fully reflect the complexity of incoherent TBox.

As for future work, we plan to improve our benchmark under realistic semantic web conditions to evaluate reasoners by using realistic TBox data, and focus on different ontology reasoning and debugging algorithms to evaluate their completeness and correctness by using our extended benchmark.

References

1. Sirin Evren, Parsia Bijan, et al., Pellet:A practical OWL-DL reasoner, Web Semantics: science, services and agents on the World Wide Web, 2007, 5(2):51-53.

2. Rob Shearer, Boris Motik, and Ian Horrocks, HermiT: A highly-efficient owl reasoner, in: Proceedings of the 5th International Workshop on OWL: Experiences and Directions. Karlsruhe, Germany. 2008.
3. Dmitry Tsarkov and Ian Horrocks, FaCT++ Description Logic Reasoner: System Description, in: Proceddings of Third International Joint Conference on Automated Reasoning. Seattle, WA, USA, 2006, pp.292-297.
4. Edward Thomas, Jeff Z. Pan, Yuan Ren, TrOWL: Tractable OWL 2 Reasoning Infrastructure, in: Proceedings of 7th Extended Semantic Web Conference. Heraklion, Crete, Greece, 2010, pp.431-435.
5. Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin, LUBM: A benchmark for OWL knowledge base systems, Web Semantics: Science, Services and Agents on the World Wide Web, 2005, 3(2): 158-182.
6. Li Ma, Yang Yang, Zhaoming Qiu, et al., Towards a complete owl ontology benchmark, in: Proceedings of the 3rd European Semantic Web Conference (ESWC), Budva, Montenegro, June, 2006, pp.125-139.
7. Yingjie Li, Yang Yu and Jeff Hefli, A multi-ontology synthetic benchmark for the semantic web, in: Proceedings of the 1st International Workshop on Evaluation of Semantic Technologies, Shanghai, China. 2010.
8. Alfio Ferrara, Stefano Montanelli, et al., Benchmarking matching applications on the semantic web, The Semanic Web: Research and Applications, Springer Berlin Heidelberg, 2011, pp.108-122.
9. Christian Meilicke, Raul Garca-Castro, et al., MultiFarm: A benchmark for multilingual ontology matching, Web Semantics: Science, Services and Agents on the World Wide Web, 2012, 15: 62-68.
10. Maria Rosoiu, Cassia Trojahn Dos Santos, et al., Ontology matching benchmarks: generation and evaluation, in: Proceedings of the 6th ISWC workshop on ontology matching (OM), 2011, pp.73-84.
11. Zhengxiang Pan, Benchmarking DL Reasoners Using Realistic Ontologies, in: Proceedings of the OWLED05 Workshop on OWL: Experiences and Directions, Galway, Ireland, November, 2005, 188.
12. Tom Gardiner, Ian Horrocks and Dmitry Tsarkov, Automated benchmarking of description logic reasoners, in: Proceedings of the 19th International Workshop on Description Logics, Windermere, Lake District, UK, May, 2006, pp. 167-174.
13. Jurgen Bock, Peter Haase, et al., Benchmarking OWL reasoners, in: Proceedings of the ARea 2008 Workshop, Tenerife, Spain, June, 2008.
14. Samantha Bail, Bijan Parsia, Ulrike Sattler, JustBench: a framework for OWL benchmarking, in: Proceedings of the 9th International Semantic Web Conference (ISWC), Shanghai, China, November, 2010, pp.32-47.
15. Yong-Bin Kang, Yuan-Fang Li, Shonali Krishnaswamy, Predicting reasoning performance using ontology metrics, in: Proceedings of the 11th International Semantic Web Conference (ISWC), Boston, MA, USA, November, 2012, pp.198-214.
16. Ji Qiu, Gao Zhiqiang, Huang Zhisheng, et al., Measuring effectiveness of ontology debugging systems, Knowledge-Based Systems, 2014, 71: 169-186.
17. Kathrin Dentler, Ronald Cornet, et al., Comparison of reasoners for large ontologies in the OWL 2 EL profile, Semantic Web, 2011, 2(2): 71-87.
18. Stefan Schlobach, Zhisheng Huang, Ronald Cornet, et al., Debugging incoherent terminologies, Journal of Automated Reasoning, 2007, 39(3):317-349.
19. Aditya Kalyanpur, Debugging and repair of owl ontologies, Washington DC, American: The University of Maryland, 2006.