Proceedings of the KI 2015 Workshop on Formal and Cognitive Reasoning

5th Workshop on Dynamics of Knowledge and Belief (DKB-2015) and 4th Workshop KI & Kognition (KIK-2015)

Christoph Beierle, Gabriele Kern-Isberner,
Marco Ragni, Frieder Stolzenburg (eds.)

Dresden, 22-Sep-2015

Contents

Preface

Information for real life AI applications is usually pervaded by uncertainty and subject to change, and thus demands for non-classical reasoning approaches. At the same time, psychological findings indicate that human reasoning cannot be completely described by classical logical systems. Sources of explanations are incomplete knowledge, incorrect beliefs, or inconsistencies. Generally, people employ both inductive and deductive reasoning to arrive at beliefs; but the same argument that is inductively strong or powerful may be deductively invalid. Therefore, a wide range of reasoning mechanisms has to be considered, such as analogical or defeasible reasoning. The field of knowledge representation and reasoning offers a rich palette of methods for uncertain reasoning both to describe human reasoning and to model AI approaches. Its many facets like qualitative vs. quantitative reasoning, argumentation and negotiation in multi-agent systems, causal reasoning for action and planning, as well as nonmonotonicity and belief revision, among many others, have become very active fields of research. Beyond computational aspects, these methods aim to reflect the rich variety of human reasoning in uncertain and dynamic environments.

The aim of the series of workshops is on the one hand to address recent challenges and to present novel approaches to uncertain reasoning and belief change in their broad senses and in particular provide a forum for research work linking different paradigms of reasoning and on the other hand to foster a multidisciplinary exchange between the fields of AI and cognition by bringing together researchers from artificial intelligence, automated deduction, computer science, cognitive psychology, and philosophy. Previous events of the Workshop on *Dynamics of Knowledge and Belief* (DKB) took place in Osnabrück (2007), Paderborn (2009), Berlin (2011), and Koblenz (2013). Previous editions of the Workshop on *KI & Kognition* (KIK) took place in Saarbrücken (2012), Koblenz (2013), and Stuttgart (2014).

This year, we put a special focus on papers from both fields that provide a base for connecting formal-logical models of knowledge representation and cognitive models of reasoning, addressing formal as well as experimental or heuristic issues. Reflecting this focus, the workshop *Formal and Cognitive Reasoning* at KI 2015 is organized jointly by the GI special interest groups *FG Wissensrepräsentation und Schließen* and *FG Kognition*.

Out of eight submissions, five have been selected for presentation at the workshop after a thorough review process, four of them as regular papers and one as a short paper. In consequence, the workshop hosts contributions on learning rules for cooperative problem solving, qualitative probabilistic inference with default inheritance, algebraic semantics for graded propositions, functional completeness of argumentation semantics, and approximate reasoning with fuzzy-syllogistic systems. We are happy also to have two invited talks, jointly with the *Workshop on Neural-Cognitive Integration* (NCI @ KI 2015) and the *29th Workshop on (Constraint) Logic Programming* (WLP 2015). The two invited speakers, Herbert Jaeger and Steffen Hölldobler, both outstanding researchers in their respective fields, present interesting insights in recurrent neural networks with conceptors, and aim at combining human reasoning, logic programs and connectionist systems, respectively.

Acknowledgments

The organizers of this workshop would like to thank the organizers of the KI 2015 conference in Dresden, especially the workshop chair, Anni-Yasmin Turhan, for their excellent support. We also would like to thank the members of the program committee for their help in selecting and improving the submitted papers, and finally all participants of the workshop for their contributions. Our wish is that new inspirations and collaborations between the contributing disciplines will emerge from this workshop.

Christoph Beierle, Gabriele Kern-Isberner, Marco Ragni, Frieder Stolzenburg

Workshop Organizers and Co-Chairs

Christoph Beierle	FernUniversität in Hagen, Germany
Gabriele Kern-Isberner	TU Dortmund, Germany
Marco Ragni	Universität Freiburg, Germany
Frieder Stolzenburg	Hochschule Harz, Germany

Program Committee

Thomas Barkowsky	Universität Bremen, Germany
Gerd Brewka	Universität Leipzig, Germany
Igor Douven	Paris-Sorbonne University, France
Christian Freksa	Universität Bremen, Germany
Ulrich Furbach	Universität Koblenz, Germany
Joachim Hertzberg	Universität Osnabrück, Germany
Andreas Herzig	Universite Paul Sabatier, Toulouse, France
Steffen Hölldobler	Technische Universität Dresden, Germany
Manfred Kerber	University of Birmingham, UK
Gerhard Lakemeyer	RWTH Aachen, Germany
Bernhard Nebel	Albert-Ludwigs-Universität Freiburg, Germany
Henri Prade	IRIT – CNRS, France
Hans Rott	Universität Regensburg, Germany
Ute Schmid	Universität Bamberg, Germany
Matthias Thimm	Universität Koblenz-Landau, Germany
Paul Thorn	Universität Düsseldorf, Germany
Hans Tompits	TU Wien, Austria
Christoph Wernhard	Technische Universität Dresden, Germany
Stefan Wölfl	Albert-Ludwigs-Universität Freiburg, Germany

Shaping the Dynamics of Recurrent Neural Networks by Conceptors

Herbert Jaeger

Jacobs University Bremen
Campus Ring
28759 Bremen
Germany

Summary

The human brain is a dynamical system whose extremely complex sensor-driven neural processes give rise to conceptual, logical cognition. Understanding the interplay between nonlinear neural dynamics and concept-level cognition remains a major scientific challenge. Here I propose a mechanism of neurodynamical organization, called conceptors, which unites nonlinear dynamics with basic principles of conceptual abstraction and logic. It becomes possible to learn, store, abstract, focus, morph, generalize, de-noise and recognize a large number of dynamical patterns within a single neural system; novel patterns can be added without interfering with previously acquired ones; neural noise is automatically filtered. Conceptors may help to explain how conceptual-level information processing emerges naturally and robustly in neural systems, and may help to remove a number of roadblocks in the theory and applications of recurrent neural networks.

Human Reasoning, Logic Programs and Connectionist Systems

Steffen Hölldobler

International Center for Computational Logic
Technische Universität Dresden
01062 Dresden
Germany

Summary

The suppression task, the selection task, the belief bias effect, spatial reasoning and reasoning about conditionals are just some examples of human reasoning tasks which have received a lot of attention in the field of cognitive science and which cannot be adequately modeled using classical two-valued logic. I will present an approach using logic programs, weak completion, three-valued Łukasiewicz logic, abduction and revision to model these tasks. In this setting, logic programs admit a least model and reasoning is performed with respect to these least models. For a given program, the least model can be computed as the least fixed point of an appropriate semantic operator and, by adapting the Core-method, can be computed by a recurrent connectionist network with a feed-forward core.

Learning Rules for Cooperative Solving of Spatio-Temporal Problems

Daan Apeldoorn

Information Engineering Group,
Technische Universität Dortmund, Germany
daan.apeldoorn@tu-dortmund.de

Abstract. This paper addresses the issue of creating agents that are able to learn rules for cooperative problem solving behavior in different multi-agent scenarios. The proposed agents start with given rule fragments that are combined to rules determining their action selection. The agents learn how to apply these rules adequately by collecting rewards retrieved from the simulation environment of the scenarios. To evaluate the approach, the resulting agents are applied in two different example scenarios.

Keywords: multi-agent simulation, cooperative problem solving, rule learning, knowledge extraction

1 Introduction

Learning to solve problems cooperatively is an interesting and challenging task for agents in multi-agent scenarios with possible applications in logistics, scheduling or robotics (e. g., [5]). In this paper, it is tried to approach this issue by introducing agents that learn to apply rules which are combined from given rule fragments. The agents are then evaluated in the context of different spatio-temporal problem solving scenarios that are defined using the multi-agent framework AbstractSwarm [1, 2] and the learned rules are extracted from the agents' epistemic state.

Section 2 introduces the agent model. Section 3 evaluates the agent model in the context of two different example scenarios and the results are presented. A conclusion and an outlook on future work are given in Section 4.

2 Agent Model

The agent model is considered a template which is instantiated for every agent that is part of a multi-agent scenario. Thus, only homogeneous multi-agent scenarios are considered here, where every agent follows the same implementation. Agents have a *collective* epistemic state, but every agent infers its own actions also considering its current percepts.

In AbstractSwarm the two basic concepts *agents* and *stations* exist. The former refer to the active components of a scenario (being able to act) and the latter represent the passive components of a scenario (serving as *locations* of the agents). These concepts are used in the following as a basis for the agents' percepts and actions.

2.1 Percepts, Actions and Communication

Following the agent interface of the AbstractSwarm simulation framework, the percepts of an agent a consist of:

- a's current location (i. e., the station where a is currently located)
- the current location of all other agents (i. e., the stations where the other agents are currently located)
- the next action of every agent $a_i \neq a$ that was computed in the same time step *before* a (i. e., the station every agent a_i selected as its next target)[1]

An agent's action is defined as the decision about its next target location chosen from a given set of potential target locations.

Before the action of an agent a is executed (i. e., after the agent decides about its next target location, but before the agent starts moving towards this new target location), the agent communicates its decision to all other agents $a_j \neq a$ that are computed *subsequently* to a. This information then is part of the percepts of the agents a_j (see the third point at the beginning of Section 2.1).

2.2 Rules

The agents are provided with a set of *rule fragments* that can be combined to *rules*. The resulting rules are used by the agents to select their target locations. A rule fragment is either the *subject* or the *strategy* of a rule. A rule subject represents a property of a location (e. g., the current free space of the location) and a rule strategy is a selection criterion (e. g., whether a location should be selected according to the maximum or the minimum value of a given subject). A complete rule is of the form *subject_strategy*, stating that locations will be selected based on the given subject and according to the given strategy.

As an Example, consider the rule *FreeSpace_Max*: Following this rule, an agent would choose, among all potential target location, the one with the maximum free space as next target.

Since locations can have a broad variety of different properties that could potentially serve as selection criteria, a subset of rule subjects must be selected here. The focus is set to subjects that seem to be relevant for the example scenarios considered in Section 4:[2]

[1] Note that in the AbstractSwarm framework, in every discrete time step, all agents of a scenario are computed subsequently (in random order) and agents can consider the information communicated by other agents that were computed before.

[2] Note that the selection of possible rules is also restricted by the limited number of properties that are available in the AbstractSwarm framework.

- *FreeSpace*: The current available space of a location (e. g., the remaining number of car agents that are still able to enter a parking deck until it will be completely covered)
- *MaxSpace*: The total available space of a location (e. g., the total number of parking boxes of a parking deck)
- *VisitTime*: The duration of visiting a location, excluding the time to get to the location (e. g., the time that will be needed to do a transaction on a bank counter)
- *RemainingTime*: The time left until a fully covered location becomes available again (e. g., the remaining time a bank counter will be occupied while a bank customer agent is finishing its transaction)

The following rule strategies are selected, which are covering the most common cases:

- *Min*: Selection according to the minimal value of the rule subject
- *Max*: Selection according to the maximal value of the rule subject
- *Avg*: Selection according to the average value of the rule subject
- *None*: No selection according to the value of the rule subject (random selection regarding the rule subject)

In case the application of a rule results in the selection of more than one target location (i. e., the values of the rule subject are equal), multiple rules can be chained, meaning that the rules are applied subsequently.

As an example, following the rule chain *FreeSpace_Max* → *VisitTime_Min*, agents will first select the locations with a maximum amount of free space (compared to all other potential target locations). If more than one location are currently having the same amount of free space, the one with the minimum visiting time will be selected from these.

If all rules are chained and there are still more than one location in the result set, one of the locations in the result set is chosen randomly.

The order of the rule chains is learned based on the agents' experiences and is inferred from the agents' current epistemic state. The epistemic state together with the learning and the inference process will be explained in the following.

2.3 Epistemic State

The epistemic state of an agent comprises its knowledge about which rules (or rather rule chains) are most preferable. Since the usefulness of rules strongly depends on the problem to be solved (and in many cases there is no a priori knowledge about which rules could be useful), the agents learn their epistemic state by acting in the problem scenario and by earning rewards for their actions. The epistemic state is represented as a Bayesian Logic Network [3] which is trained using Statistical Relational Learning. By this, the learned rules can later be extracted easily from the epistemic state.

In the following three subsections, the Bayesian Logic Network representing the epistemic state will be introduced and both the learning and the inference process will be described.

Bayesian Logic Network. A Bayesian Logic Network (BLN), introduced by Jain et al. in [3], is a Bayesian Network extended by (many-sorted) first-order Logic. Functions and predicates (which are considered functions with the co-domain {True, False}) are associated with the nodes of an underlying Bayesian Network, such that every node represents a (conditional) probability distribution over the co-domain of the associated predicate. Logical formulas can be added, which then will be considered by the inference mechanism (in addition to the probability distributions of the network). A BLN can be trained such that the (conditional) probability distributions are learned from a given set of data. The BLN is implemented and visualized in the following using ProbCog [4], a software suite for Statistical Relational Learning.

Besides some logical formulas, the BLN for modeling the epistemic state of the agents consists of only three nodes:

- The node *selectedValue(sit)* represents the function with the corresponding probability distribution for the rule subjects.
- The node *applyRule(sit)* represents the function with the corresponding conditional probability distribution for the rule strategies, given a rule subject. The associated function of the node is later queried for inference.
- The isolated node *valueComplete(sit, val)* represents a predicate which is only relevant for the logical formulas implemented in the BLN. (These formulas are used later to handle special cases where some of the potential target locations of an agent are lacking a property and therefore are incomparable regarding this subject, see explanation of the logical formulas below).

The functions associated with the nodes *selectedValue(sit)* and *applyRule(sit)* depend on the current *situation* of an agent (which is used to specify evident knowledge for the inference queries and for the training data later). The predicate *valueComplete(sit, val)* depends on the current situation of the agent and on the subject of a rule. Figure 1 shows the described BLN in the initial state.

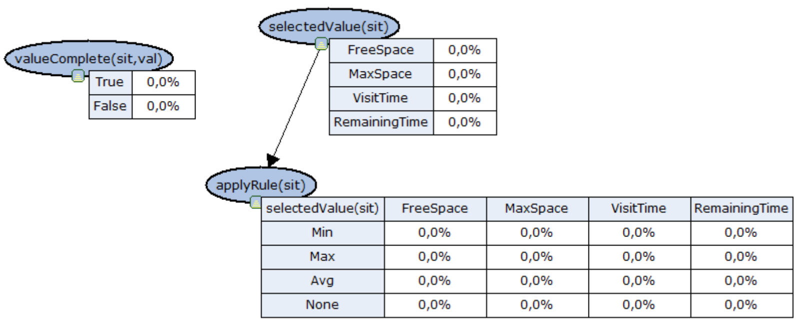


Fig. 1. Initial BLN for the agents' epistemic state.

Note that the agents are using a *collective* epistemic state (see Section 2), such that all agents access the same BLN (both for learning and inference).

However, every agent a infers its own actions, which are additionally depending on its current external state σ_t^a at time t in the environment.

The logical rules implemented in the BLN handle the cases where some locations are lacking some of the properties represented by the rule subjects, which can lead to anomalies for certain subjects (e. g., a closed bank counter does not have a limited duration for visiting, thus applying the rule *VisitTime_Max* would always lead to the selection of the closed bank counter). These kinds of problems are covered by the following formulas:

$$selectedValue(sit, VisitTime) \wedge \neg valueComplete(sit, VisitTime) \\ \Rightarrow \neg applyRule(sit, Max)$$

$$selectedValue(sit, RemainingTime) \wedge \neg valueComplete(sit, RemainingTime) \\ \Rightarrow \neg applyRule(sit, Max)$$

The first formula states that the rule *VisitTime_Max* is never applied in case not all of the potential target locations have a defined duration for visiting. The second formula states this analogously for the time left until a fully covered location is available again.

Learning. While a simulation episode is running, the agents select their locations by trying out different rule subjects and strategies. After the execution of an action is completed by an agent $a \in A$ (i. e., after the agent visited a selected location), the agent gets a *local* reward r from the simulation environment, which is calculated as follows:

$$r := \frac{1}{t_{rwd} - t_{act} + 1} \sum_{t=t_{act}}^{t_{rwd}} \left(\frac{1}{|A|} \sum_{a \in A} w(\sigma_t^a) \right) \tag{1}$$

where t_{act} is the point in time when a selected a target location, t_{rwd} is the point in time when a finished visiting this location, σ_t^a describes the state of a at time t, and function w is defined as:

$$w(\sigma_t^a) := \begin{cases} 1, \text{ if } a \text{ is visiting a location at time } t \\ 0, \text{ otherwise} \end{cases} . \tag{2}$$

Thus, the reward for the action of agent a is calculated from the number of agents in the scenario, that are visiting locations as a consequence of the action performed by a, averaged over time t_{act} to time t_{rwd}: Agents gain higher rewards, the more their actions allow other agents to simultaneously perform their respective actions and lower rewards the more they are restricting other agents.

An agent can easily calculate a *global* reward from the local rewards of its actions by summing up all local rewards until the end of a simulation episode.

Before a new simulation episode starts, the agents store their experiences from the previous episode (i. e., which rules were applied in which situations) by logging training datasets that consist of value assignments to the functions. The following example shows three exemplary training data records:

$$selectedValue(sit_1) = FreeSpace$$
$$applyRule(sit_1) = Max$$

$$selectedValue(sit_2) = FreeSpace$$
$$applyRule(sit_2) = Max$$

$$selectedValue(sit_3) = VisitTime$$
$$applyRule(sit_3) = Min$$

A situation identifier sit_i contains the name of the agent that was applying a rule and an index value (including the time when the rule was applied). In the given example, it was preferable in two cases to select a target location according to the rule *FreeSpace_Max* and in only one case it was preferable to select a target location according to the rule *VisitTime_Min*.

The amount of records that are stored depends on the global reward earned by the agent during an episode: The higher the reward, the more training data records are stored. After storing the training data records, the (conditional) probability distributions of the BLN are updated by counting relative frequencies from the training data.

Inference. For inferring results from the trained BLN, the function associated with the node *applyRule(sit)* is queried for every rule subject (i. e., for every value of the co-domain of the function associated with the node *selectedValue(sit)*). The results are the conditional probabilities over the different rule strategies, given the subjects.

Based on the conditional probabilities retrieved from the BLN, if an agent determines its next target location, it performs the following steps:

1. The rule (i. e., the subject-strategy-combination) with the overall highest probability value is applied to select a target location.
2. If this results in more than one location (i. e., the resulting locations are indistinguishable regarding the rule subject), the rule with the next lower probability value is applied to the results from the previous rule. This is continued until there is only one location left in the result set or until all rule subjects were used. Thus, the conditional probabilities $P(Str_1|Sub_1) > ... > P(Str_n|Sub_n)$ would lead to the rule chain $Sub_1\_Str_1 \rightarrow\ ...\ \rightarrow Sub_n\_Str_n$. (If in this case there are still more than one location in the result set, one of the remaining locations is selected randomly.)

By this, stronger rules, that where reinforced through the learning process, are preferred over weaker rules and weaker rules are used with lower priority in a rule chain (in case not all stations could be distinguished by the stronger rules).

To explore the different combinations of rule fragments (even if the agents already learned that some rules are preferable), an exploration probability determines in how many cases an agent decides to use another rule than the one that was inferred from its current epistemic state. The exploration probability depends on the number of already tried rules and the total amount of change in the conditional probability distribution. By this, the exploration probability is slightly discounted over time with increasing experience of the agent.

3 Evaluation

3.1 Test Scenarios

This section introduces the two example scenarios from [1], which are used here as test scenarios to evaluate the agent model. In both scenarios, agents have to solve a problem cooperatively. The scenarios are modeled using the AbstractSwarm framework.

Scenario 1: School Timetabling. In this scenario, a small fictive school is considered, where timetables have to be created for teachers, pupils and rooms. The school comprises:

- 2 math teachers, 2 English teachers and 1 music teacher
- 5 classes (every class has to get 2 math lessons, 2 English lessons, 1 music lesson)
- 4 rooms (3 normal class rooms, 1 special music room)

Teacher agents, class agents and course agents must organize themselves to efficiently create an optimized timetable for the school (i. e., which agent has to be at which time in which room). In this scenario, all locations have the same size (e. g., only one class can be located in a room at a point in time) and the duration of visiting is identical for all locations (i. e., all courses have the same length).

Scenario 2: Production Simulation. In this scenario, a small factory is considered, where workers are producing different products. As part of the quality assurance process, the products have to be analyzed using different machines. The factory comprises:

- 8 workers
- 2 kinds of products (5 of each kind, one kind having the need to be analyzed at higher priority)
- 3 machines (2 of which being able to analyze on their own, 1 needing a worker to monitor the analysis

Worker agents and product agents must organize themselves to efficiently create an optimized production plan with few waiting times on the machines. Machines are considered locations with different amounts of space (i. e., one of the machines is able to analyze more than one product at a time) and with different durations of the production and the analysis processes (i. e., the production times depend on the kinds of products and the analysis time needed depends on the specific kind of analysis of a machine.)

3.2 Results

First, the BLNs learned by the agents are inspected in this section and the learned rules (or rule chains) are extracted from the BLNs. After that, the overall performance of the agents for finding adequate solutions for the scenarios will be analyzed.

Learned Rules. In both cases, the agents started without any a priori knowledge about the (conditional) probabilities of the rule fragments (as shown in Figure 1) and 100 runs were performed. The resulting BLNs are shown in Figure 2 and Figure 3.

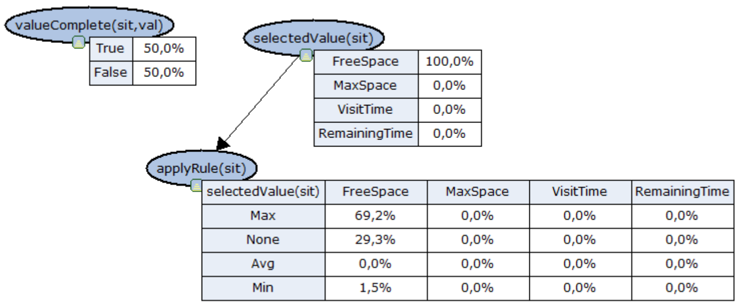


Fig. 2. Learned BLN for School Timetabling (Scenario 1) after 100 runs.

From the BLN for Scenario 1 (Figure 2) it can be seen that the agents learned the rule *FreeSpace_Max* with a high success probability. Since in Scenario 1 all locations are of the same size and all courses have the same duration, no further distinctions can be made regarding other rule subjects. Thus, this is the only rule that could be learned.

In case of Scenario 2 (Figure 3), it can be extracted from the BLN, that the rule chain *FreeSpace_Max*→*VisitTime_Avg*→*RemainingTime_Min* was learned (since $P(Max|FreeSpace) > P(Avg|VisitTime) > P(Min|RemainingTime)$). Thus, like in case of Scenario 1, it also seems to be useful here to select a target location according to its current available space. But the result is less clear than in Scenario 1: As second and third criteria, agents select their locations according to the average duration of a location (i. e., the average duration of production

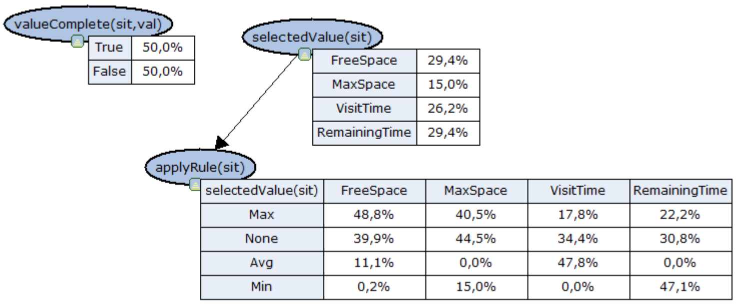


Fig. 3. Learned BLN for Production Simulation (Scenario 2) after 100 runs.

and analysis tasks) and according to the minimal time left until a fully covered production or analysis location becomes available again.

Performance. To analyze the performance of the learning agents, the *total waiting time* of all agents after solving a scenario is considered (i. e., the sum of the idle times of every agent, after all tasks defined in the scenario description were completed). Therefore, 20 repetitions of 100 runs are performed for each scenario and the results are averaged over the 20 repetitions. Every repetition is divided into two phases:

1. The first 50 runs are a *learning phase* where the exploration probability is discounted slightly depending on the experience of the agents (as described in Section 2.3).
2. The second 50 runs are an *exploitation phase*, where the exploration probability is set to zero and the agents act only based on the rules learned in the first phase.

After every repetition, the probability distributions of the BLN are reset to the initial state shown in Figure 1.

Figure 4 and Figure 5 show the performance results for the school timetabling and the production scenarios: The curves represent the minimal waiting time of all agents after r runs (averaged over the 20 repetitions). The gray bars show the waiting time of one selected representative repetition. Note that the agents do not always find a solution for a scenario: The missing gray bars (Figure 5) indicate that the agents could not find a valid solution in this simulation run fulfilling all constraints of the scenario description.

In both Figure 4 and Figure 5 it is shown that the agents are able to quickly find rather good solutions in the two scenarios. Some good solutions are already found randomly at early stages of the learning phase, but it can be seen that the overall performance is getting better through exploitation of the learned rules.

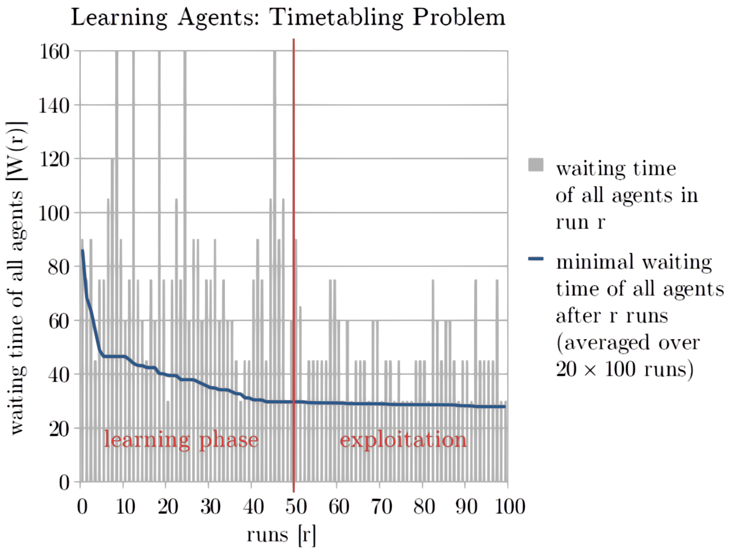


Fig. 4. Performance of learning agents in the school timetabling scenario (Scenario 1).

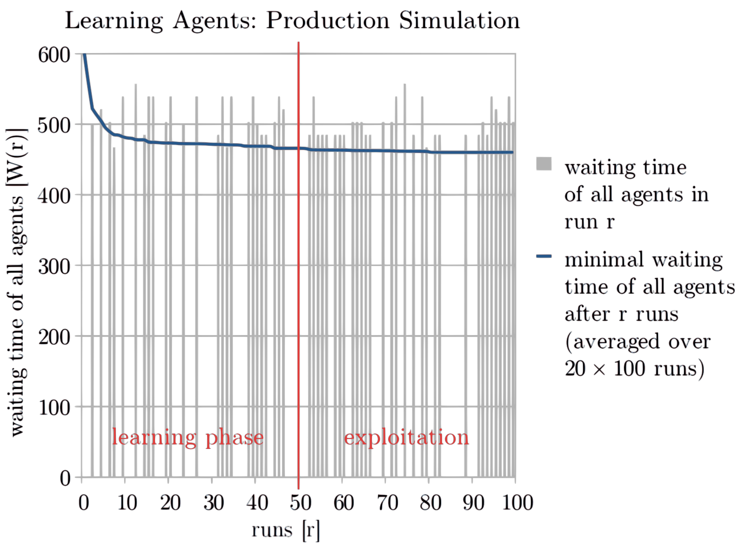


Fig. 5. Performance of learning agents in the production scenario (Scenario 2).

4 Conclusion and Future Work

In this paper, an agent model based on a BLN was presented where multiple agent instances are able to collectively learn rules (and rule chains) for cooperative solving of spatio-temporal problems.

The resulting agents were evaluated in the context of two example scenarios and the results showed that the agents benefit from applying the learned rules (and rule chains).

Unlike other agent-based learning techniques (like Reinforcement Learning), the learned knowledge (i. e., the rules and rule chains) can be easily inspected and extracted from the agents (as shown in Section 3.2) and the agents can be adapted or extended by adding further rule subjects or strategies. Besides that, learning behavioral rules rather than state-action-pairs reduces the state-action-space significantly, which is especially useful in high-dimensional environments (as it is inherently the case in multi-agent-systems, since the state-action-space grows exponentially with the number of agents [6]).

As future work, the rule learning approach could be tested in further, more open environments, as it is the case e. g., for robots cooperating in real world environments (for a related real world scenario see e. g. [5]).

References

1. Apeldoorn, D.: A spatio-temporal multiagent simulation framework for reusing agents in different kinds of scenarios. *To be published in the proceedings of the MATES 2015 conference.*
2. Apeldoorn, D.: Abstractswarm – a generic graphical modeling language for multi-agent systems. In: Klusch, M., Thimm, M., Paprzycki, M. (eds.) Multiagent System Technologies. LNCS, vol. 8076, pp. 180–192. Springer, Berlin Heidelberg (2013)
3. Jain, D., Waldherr, S., Beetz, M.: Bayesian logic networks (extended version). Technical report ias-2009-03, Technische Universität München (2009)
4. Probcog toolbox – a toolbox for statisitical relational learning and reasoning. `http://ias.in.tum.de/software/probcog`, last accessed on 2015-08-15
5. Scheuren, S., Stiene, S., Hertzberg, J., Hartanto, R., Reinecke, M.: The problem of spatio-temporally constrained motion planning for cooperative vehicles. In: Proceedings of the 26th Workshop "Planen, Scheduling und Konfigurieren, Entwerfen" (PuK 2011) (2011)
6. Tan, M.: Independent vs. cooperative agents. In: Proceedings of the Tenth International Conference on Machine Learning. pp. 330–337. Morgan Kaufmann, San Francisco (1993)

Qualitative Probabilistic Inference with Default Inheritance

Paul Thorn*,[a], Christian Eichhorn†,[b], Gabriele Kern-Isberner†, and Gerhard Schurz*

*: Institute for Philosophy, Heinrich Heine Universität Düsseldorf, Düsseldorf, Germany
†: Department of Computer Science, Technische Universität Dortmund, Dortmund, Germany
[a]:thorn@phil.hhu.de, [b]:christian.eichhorn@tu-dortmund.de

Abstract. There are numerous formal systems that allow inference of new conditionals based on a conditional knowledge base. Many of these systems have been analysed theoretically and some have been tested against human reasoning in psychological studies, but experiments evaluating the performance of such systems are rare. In this article, we extend the experiments in [19] in order to evaluate the inferential properties of c-representations in comparison to the well-known Systems P and Z. Since it is known that System Z and c-representations mainly differ in the sorts of inheritance inferences they allow, we discuss subclass inheritance and present experimental data for this type of inference in particular.

1 Introduction

There are systems of conditional reasoning (such as Adams' System P [2]) that can be used to make *valid* (i.e., truth preserving) inferences about conditional probabilities. More generally, there are systems of conditional reasoning where it is plausible to adopt a *probabilistic interpretation* of conditionals, where conditionals of the form $(\psi|\phi)$ are *interpreted* as expressing that the corresponding conditional probability, $P(\psi|\phi)$, is high. In some cases, it may be plausible to adopt a *probabilistic interpretation* of conditionals, for a given system, even when the inferences licensed by the respective system are ampliative, and not truth preserving, given the probabilistic interpretation. For example, although inheritance inference (i.e., from $(\psi|\phi)$ infer $(\psi|\phi \wedge \chi)$) may fail to preserve high probability in many cases, inheritance inference is a reasonable form of inference that one might like to codify within a system of conditional reasoning.

In the present paper, we compare and evaluate the behaviour of two systems of conditional reasoning that are stronger than System P, but admit of a probabilistic interpretation, namely: System Z [16], and System MinC (which we define based on the inductive method of c-representations [8,9]). The two systems are of interest, since they both license a number of desirable inference patterns, such as inheritance inference and contraposition, that are not licensed by System P. Nevertheless the two systems differ in some important respects, such as in their treatment of inheritance reasoning.

Within a system where conditionals are treated as expressing defaults, it is desirable that *subclass inheritance* among defaults be licensed, defeasibly. For example, from the default that birds usually can fly we would like to infer that crows (a subclass of birds) are usually capable of flight, in the case where we have no background knowledge indicating that crows are exceptional birds. Such inferences are assumed to be defeasible, meaning that there are conditions under which such inferences are defeated (i.e., conditions under which the inference is not licensed).

Beyond defeasible subclass inheritance, it is controversial whether *inheritance in the case of exceptional subclasses* should be licensed, defeasibly [6]. For example, notice that penguins are exceptional birds inasmuch as they lack the capacity of flight. Given that penguins represent an exceptional subclass of the class of birds, it is controversial whether the subclass, penguins, should inherit other characteristics typical of birds. For example, assuming that birds usually have wings, it is controversial whether it is reasonable to infer that penguins usually have wings, given that they are (usually) incapable of flight.

A principal difference between System Z and System MinC is that the latter, and not the former, permits inheritance inference in the case of exceptional subclasses. Prima Facie, this fact speaks in favor of System MinC, a point which we briefly discuss in Section 5. However, as our primary means of evaluation, our paper reports the results of experiments which test the behaviour of Systems Z and MinC in reasoning about a simulated stochastic environment. For additional perspective, we also tested the behavior of System P and System QC [19,23]. The results show that while System MinC makes *many* inferences that are not drawn by System Z, System Z *rarely* makes an inference that is not drawn by System MinC. Since the two systems are both ampliative with respect to the *probabilistic interpretation* of conditionals (in contrast to System P), it is clear that the conclusions drawn by Systems Z and MinC are more risky than the ones drawn by System P. It is also plausible to think that conclusions that are drawn by System MinC and not System Z are more *risky* than the conclusions that are drawn by both systems, since such conclusions go "farther out on a limb". The results presented here vindicate this thought, and provide a clearer picture of just how risky these inferences are.

The paper is organised as follows: After introducing the necessary formal preliminaries in Section 2, we introduce Systems P, Z, and QC in Section 3. We define System MinC via c-representations in Section 4. In Section 5 we discuss subclass inheritance for exceptional subclasses. We present the experimental setup and the results of the experiments in Sections 6 and 7, and conclude in Section 8.

2 Preliminaries

Let $\Sigma = \{V_1, ..., V_m\}$ be a propositional alphabet where a *literal* is a variable V interpreted to $true$ (v) or $false$ ($\overline{v}$). From these we obtain the propositional language $\mathfrak{L}$ as the set of formulas of Σ closed under negation $\neg$, conjunction $\wedge$, and disjunction $\vee$, as usual; for shorter formulas, we abbreviate conjunction by juxtaposition (i.e., ab is equivalent to $a \wedge b$), and negation by overlining (i.e., $\overline{a}$ is equivalent to $\neg a$). We write the material implication as $\phi \rightarrow \psi$ which is, as usual, equivalent to $\overline{\phi} \vee \psi$. *Interpretations* or *possible worlds* are also defined in the usual way; the set of all possible worlds is denoted by Ω. We often take advantage of the 1-1 association between worlds and *complete conjunctions*, i.e., conjunctions of literals where every $V_i \in \Sigma$ appears exactly once.

Table 1. Evaluation of conditionals in the penguin example ($+$ indicates verification, $-$ falsification, an empty cell inapplicability) (above). Two OCF for the penguin example (below).

	$pbfw$	$pbf\overline{w}$	$pb\overline{f}w$	$pb\overline{f}\overline{w}$	$p\overline{b}fw$	$p\overline{b}f\overline{w}$	$p\overline{b}\,\overline{f}w$	$p\overline{b}\,\overline{f}\overline{w}$	$\overline{p}bfw$	$\overline{p}bf\overline{w}$	$\overline{p}b\overline{f}w$	$\overline{p}b\overline{f}\overline{w}$	$\overline{p}\overline{b}fw$	$\overline{p}\overline{b}f\overline{w}$	$\overline{p}\overline{b}\,\overline{f}w$	$\overline{p}\overline{b}\,\overline{f}\overline{w}$
$(f\|b)$	$+$	$+$	$-$	$-$					$+$	$+$	$-$	$-$				
$(\overline{f}\|p)$	$-$	$-$	$+$	$+$	$-$	$-$	$+$	$+$								
$(b\|p)$	$+$	$+$	$+$	$+$	$-$	$-$	$-$	$-$								
$(w\|b)$	$+$	$-$	$+$	$-$					$+$	$-$	$+$	$-$				
$\kappa^Z_\Delta(\omega)$	2	2	1	1	2	2	2	2	0	1	1	1	0	0	0	0
$\kappa^{c'}_\Delta(\omega)$	2	3	1	2	4	4	2	2	0	1	1	2	0	0	0	0

A *conditional* $(\psi|\phi)$, $\phi, \psi \in \mathfrak{L}$, is trivalent, with the evaluation: $(\psi|\phi)$ is *verified* iff $\omega \models \phi\psi$, $(\psi|\phi)$ is *falsified* iff $\omega \models \phi\overline{\psi}$, and $(\psi|\phi)$ is *inapplicable* iff $\omega \models \overline{\phi}$ [5,8]. A finite set of conditionals $\Delta = \{(\psi_1|\phi_1), \dots, (\psi_n|\phi_n)\}$ is called a *knowledge base*.

An *Ordinal Conditional Function* (OCF, ranking function [21,20]) is a function $\Omega \to \mathbb{N}_0 \cup \{\infty\}$ that assigns to each world an implausibility rank, such that $\kappa^{-1}(0)$, the preimage of 0, is non-empty. The rank of a formula $\psi \in \mathfrak{L}$ is the rank of the lowest ranked world that satisfies the formula, formally: $\kappa(\phi) = \min_{\omega \models \phi}\{\kappa(\phi)\}$. The rank of a conditional $(\psi|\phi)$ is defined as: $\kappa(\psi|\phi) = \kappa(\phi\psi) - \kappa(\phi)$. A ranking function *accepts* a conditional $(\psi|\phi)$ (written $\kappa \models (\psi|\phi)$) iff $\kappa(\phi\psi) < \kappa(\phi\overline{\psi})$; κ is *admissible* with respect to a knowledge base Δ if and only if $\kappa \models (\psi|\phi)$ for all $(\psi|\phi) \in \Delta$.

Example 1. We illustrate these preliminaries with the well-known penguin example. Let B indicate whether something is a bird (b) or not ($\overline{b}$), let P indicate whether something is a penguin (p) or not ($\overline{p}$), let F indicate whether something is capable of flying (f) or not ($\overline{f}$), and let W indicate whether something has wings (w) or not ($\overline{w}$). This gives us the alphabet $\Sigma = \{P, B, F, W\}$ with a set of worlds given in the top row of Table 1. We use the conditionals "birds usually can fly" $(f|b)$, "penguins usually cannot fly" $(\overline{f}|p)$, "penguins usually are birds" $(b|p)$, and "birds usually have wings" $(w|b)$ to compose the knowledge base $\Delta = \{(f|b), (\overline{f}|p), (b|p), (w|b)\}$. Table 1 displays the evaluation of these conditionals within the worlds $\omega \in \Omega$. Table 1 also displays two ranking functions, κ^Z_Δ and $\kappa^{c'}_\Delta$ that are admissible with respect to this knowledge base. We discuss the two ranking functions, especially how they are generated inductively from the above knowledge base, later in the paper.

3 Overview of Systems P, Z, and QC

As described in [7], System P represents the confluence of a number of different semantic criteria. One feature of System P that is of interest here is its connection with the following consequence relation (cf. [2]):

Improbability-Sum Preservation: $(\psi_1|\phi_1), ..., (\psi_n|\phi_n) \models_{i.s.p.} (\xi|\chi)$ iff for all probability functions, P, over the appropriate language: $I(\xi|\chi) \leq \Sigma_{i=1}^n\, I(\psi_i|\phi_i)$, where $I(\psi|\phi)$, the *improbability* of ψ given ϕ, is defined as $1 - P(\psi|\phi)$.

As Adams [2] demonstrated, the following calculus (denoted by $\vdash_P$) is correct and complete for $\models_{i.s.p.}$:

(REF) (reflexivity) $\vdash_P (\psi|\phi)$
(LLE) (left logical equivalence) if $\models (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$, then $(\chi|\phi) \vdash_P (\chi|\psi)$
(RW) (right weakening) if $\models \phi \rightarrow \psi$, then $(\phi|\chi) \vdash_P (\psi|\chi)$
(CC) (cautious cut) $(\psi|\phi), (\chi|\psi \wedge \phi) \vdash_P (\chi|\phi)$
(CM) (cautious monotony) $(\psi|\phi), (\chi|\phi) \vdash_P (\chi|\psi \wedge \phi)$
(AND) $(\phi|\chi), (\psi|\chi) \vdash_P (\phi \wedge \psi|\chi)$
(OR) $(\chi|\phi), (\chi|\psi) \vdash_P (\chi|\psi \vee \phi)$

System P also has a semantics expressible in terms of ranking functions. In particular, $\Delta \vdash_P (\psi|\phi)$ if and only if every ranking function that is *admissible* for Δ *accepts* $(\psi|\phi)$ [1,13]. Apart from being characterized by reasonable (if mininal) principles, and plausible semantic theories, empirical studies show that human reasoning makes use of the principles of System P (c.f. [17,11]), which renders the study of System P especially worthwhile.

Inference by System Z [16] is based upon the unique ranking function κ^Z_Δ, among the admissible ranking functions for (consistent) $\Delta = \{(\psi_1|\phi_1), \ldots, (\psi_n|\phi_n)\}$, that minimizes the rank of each world in the set of possible worlds Ω_Δ defined over the propositional atoms appearing in Δ. This is achieved by forming an ordered partition $(\Delta_0, ..., \Delta_m)$ of Δ, where each Δ_i is the maximal subset of $\bigcup_{j=i}^m \Delta_j$ that is tolerated by $\bigcup_{j=i}^m \Delta_j$ (where a conditional, $(\psi|\phi)$, is *tolerated* by a set of conditionals, Δ, iff $\exists\omega$: $\omega \models \phi\psi$ and $\forall(\psi_i|\phi_i) \in \Delta$: $\omega \models \phi_i \rightarrow \psi_i$). Due to maximality, such partitions are unique for every Δ. Given the respective partition, κ^Z_Δ is defined as the OCF that assigns the value 0 to a world, ω, if no elment of Δ falsified at ω, and otherwise assigns the value $i + 1$, where i is index of the rightmost element of $(\Delta_0, ..., \Delta_m)$ that contains a conditional falsified by ω. Table 1 (above) presents κ^Z_Δ for the knowledge base described in Example 1. Inference by System Z is characterized by the relation $\vdash_Z$, which is defined in terms of the conditionals *accepted* by κ^Z_Δ:

$$\Delta \vdash_Z (\psi|\phi) \qquad \text{iff} \qquad \kappa^Z_\Delta \models (\psi|\phi). \qquad \text{(System Z)}$$

By adding the rule *Monotony*, i.e., $(\psi|\phi)$ implies $(\psi|\phi \wedge \chi)$, to System Z (or merely to System P), we obtain System QC. We here follow [19], and implement System QC by reasoning with conditionals *as if* they were material implications, and define System QC as follows:

$$\Delta \vdash_{QC} (\psi|\phi) \qquad \text{iff} \qquad \{\ \phi_i \rightarrow \psi_i \mid (\psi_i|\phi_i) \in \Delta\ \} \models \phi \rightarrow \psi \qquad \text{(System QC)}$$

4 System MinC

System MinC is defined in terms of ranking functions known as c-representations [8,9]. A c-representation assigns an individual impact value $\kappa_i^- \in \mathbb{N}_0$ to each conditional $(\psi_i|\phi_i) \in \Delta$. Using these impact values, a ranking function, κ^c_Δ, is defined, where each world ω is assigned the rank $\kappa^c_\Delta(\omega)$, which is the sum of the impacts of the conditionals falsified by ω:

$$\kappa^c_\Delta(\omega) = \sum_{i:\omega\models\phi_i\overline{\psi}_i} \kappa_i^- . \qquad (1)$$

The impacts of the conditionals are chosen so that $\kappa_{\Delta}^{c} \models \Delta$, which is the case if

$$\kappa_i^- > \min_{\omega \models \psi_i \phi_i} \Big\{ \sum_{\substack{j:\omega \models \phi_j \overline{\psi}_j \\ i \neq j}} \kappa_j^- \Big\} - \min_{\omega \models \psi_i \overline{\phi}_i} \Big\{ \sum_{\substack{j:\omega \models \phi_j \overline{\psi}_j \\ i \neq j}} \kappa_j^- \Big\}. \tag{2}$$

Entailment with respect to a c-representation is defined, as usual, via the OCF κ_{Δ}^{c}.

$$\kappa_{\Delta}^{c} \models (\psi|\phi) \qquad \text{iff} \qquad \kappa_{\Delta}^{c}(\phi\psi) < \kappa_{\Delta}^{c}(\phi\overline{\psi}). \tag{3}$$

Example 2. We illustrate c-representations with the penguin example (Example 1). Table 1 shows the verification/falsification behaviour of the worlds and the conditionals in this example, where (2) gives us:

$$\begin{aligned}
\kappa_1^- &= \min\{\kappa_2^-, \kappa_2^- + \kappa_4^-, 0, \kappa_4^-\} - \min\{0, \kappa_4^-, 0, \kappa_4^-\} \\
\kappa_2^- &= \min\{\kappa_1^-, \kappa_1^- + \kappa_4^-, \kappa_3^-, \kappa_3^-\} - \min\{0, \kappa_4^-, \kappa_3^-, \kappa_3^-\} \\
\kappa_2^- &= \min\{\kappa_2^-, \kappa_2^- + \kappa_4^-, \kappa_1^-, \kappa_1^- + \kappa_4^-\} - \min\{\kappa_2^-, \kappa_2^-, 0, 0\} \\
\kappa_1^- &= \min\{\kappa_2^-, \kappa_1^-, 0, \kappa_1^-\} - \min\{\kappa_2^-, \kappa_1^-, 0, \kappa_1^-\}
\end{aligned}$$

This can be solved via the minimal solution $\kappa_1^- = 1$, $\kappa_2^- = 2$, $\kappa_3^- = 2$, $\kappa_4^- = 1$, which, with (1) gives us the c-representation $\kappa_{\Delta}^{c'}$ shown in Table 1.

The defining system (2) is a system of *inequalities*. The system defines a schema for all c-representations of a given knowledge base Δ rather than a unique ranking function for Δ. To apply the method of c-representations to define a system of conditional infererence, we introduced an algorythm for selecting a unique c-representation for each knowledge base. We call the resulting system "*MinC*" (<u>min</u>imal <u>c</u>-representation). Following the idea of System Z being the pareto-minimal ranking function admissible to a knowledge base Δ, we define System MinC via a *minimal* c-representation that assigns the smallest possible rank to each world. Since there are no straightforward criteria for identifying a *unique* minimal c-representation, we opted for the following hierarchy of criteria (cf. [15]):

(a) minimising the combined rank $\sum_{\omega \in \Omega} \kappa(\omega)$,
(b) minimising the maximal rank $\max_{\omega \in \Omega}\{\kappa(\omega)\}$,
(c) minimising the combined impacts $\sum_{i=1}^{n} \kappa_i^-$, and
(d) minimising the maximal impact $\max_{1 \leq i \leq n}\{\kappa_i^-\}$.

Most of the time, these criteria, in this order, select a minimal c-representation within one or two steps.

To determine our designated minimal c-representation, we order c-representations by (a), the ones indistinguishable by (a) are then ordered by (b), the ones indistinguishable by (b) are then ordered by (c), followed by (d). Since ordering by (a) through (d) does not always yield a unique minimal c-representation, we implemented a practical measure for identifying our designated c-representation as that c-representation having the lexicographically smallest vector $(\kappa_1^-, ..., \kappa_n^-)$ among the minimal solutions ordered by (a) though (d). To distinguish this uinque c-representation from the general κ_{Δ}^{c}, we call the c-representation that is chosen according to the preceding tests, for respective Δ, $\kappa^{c'}_{\Delta}$, and define a corresponding inference system as follows:

$$\Delta \vdash_{MinC} (\psi|\phi) \qquad \text{iff} \qquad \kappa^{c'}_{\Delta} \models (\psi|\phi). \qquad \text{(System MinC)}$$

Note that while their is no known axiomatic characterization of $\vdash_{MinC}$ or $\vdash_Z$, both satisfy all of the principles that characterize $\vdash_P$. In addition, both $\vdash_{MinC}$ and $\vdash_Z$ satisfy *rational monotony* [12]: from $(\psi|\phi)$ and the non-validity of $(\overline{\chi}|\phi)$ infer $(\psi|\phi\chi)$.

5 Exceptionality and subclass inheritance

A principal difference between System Z and System MinC is that the latter, and not the former, permits inheritance inference in the case of exceptional subclasses. This fact is illustrated by the ranking functions $\kappa^{Z}_{\Delta}(\omega)$ and $\kappa^{c'}_{\Delta}(\omega)$ of Table 1, concering Example 1. In this case, System minC permits the conclusion that $(w|p)$, whiles System Z does not. Prima Facie, this behavior speaks in favor of System MinC. Indeed, the range of possible inheritance inferences to exceptional subclasses is very broad – broader than generally recognized – and encompasses many inferences that are generally, and correctly, regarded as reasonable. As a consequence, it appears that abandoning inheritance inference to exceptional subclasses, as a default, would forsake too much, i.e., too many reasonable inferences. Systems that do abandon these inferences are described of having a *Drowning Problem* [4].

The fact that a prohibition of inheritance inference to exceptional subclasses would forsake too much can be seen by considering a range of typical inheritance inferences, where the relevant subclass represents a small proportion of the respective superclass. For example, suppose it is given that $(f|b)$ (birds are usually able to fly), and we would like to infer $(f|jb)$ (j-birds are usually able to fly). Assume that we possess no special information regarding the class j, save that j corresponds to a relatively small (or improbable) subclass of b. In that case, we are in a position to conclude that j is exceptional relative to b, since we are in a position to accept $(\overline{j}|b)$. But it is clear that the proposed inference should be permitted. Indeed, the proposed inference is no less reasonable than the most reasonable instances of inheritance reasoning. Moreover, the fact that j corresponds to a small subclass of b does not speak against the inference. The latter point is particularly important when we consider cases of classical direct inference, where inheritance reasoning is used in order to draw a conclusion about a particular individual (see [18,3]).

6 Experiments

We here extend the experiments conducted in [19], with the aim of evaluating the performance of System MinC in comparison to System Z. To make the search space manageable, we restricted the experiments to an alphabet $\Sigma = \{A, B, C, D\}$ with a language $\mathfrak{L}^{\wedge}$ restricted to conjunctions of literals. The language of conditionals $(\mathfrak{L}^{\wedge}|\mathfrak{L}^{\wedge})$ is further restricted so that no variable may appear in both the antecedent and the consequent of a conditional. This means that $(b|a)$ and $(cd|a\bar{b})$ are in $(\mathfrak{L}^\wedge|\mathfrak{L}^\wedge)$, but $(bcd|ab)$ is not.

To generate a stochastic environment, we randomly assigned values from the real-valued interval $[0,1]$ to the probabilities: $(a|\top)$,$(b|\dot{a})$, $(c|\dot{a}\dot{b})$, and $(d|\dot{a}\dot{b}\dot{c})$ with $\dot{a} \in \{a,\overline{a}\}$, $\dot{b} \in \{b,\overline{b}\}$, and $\dot{c} \in \{c,\overline{c}\}$. We then generated the probability distribution $P : \Omega \to [0,1]$ by the so called "chain rule". Based on this distribution, *four* conditionals $(\psi|\phi)$ with $P(\psi|\phi) \geq mp$ (the *minimum probability* of the conditionals in the knowledge base for the respective simulation) were chosen randomly from $(\mathfrak{L}^\wedge|\mathfrak{L}^\wedge)$. Given this knowledge base Δ, the sets of all entailed conditionals $\mathcal{C}^X(\Delta) = \{(\psi|\phi)|(\psi|\phi) \in (\mathfrak{L}^\wedge|\mathfrak{L}^\wedge), \Delta \vdash_X (\psi|\phi)\}$, for $X \in \{\mathrm{P}, \mathrm{Z}, \mathrm{MinC}, \mathrm{QC}\}$, were computed. The restriction of our simulations to cases where the systems are provided with four premise conditionals expressed within $(\mathfrak{L}^\wedge|\mathfrak{L}^\wedge)$ partly limits the scope of our results. For some explanation concerning why these limitations are not so significant, see [24].

The accuracy of the inferences drawn by the four systems was assessed by treating the systems as asserting that the probability of the inferred conditional was at least the sum of the improbabilities of the premises upon which the inference was based. This amounts to treating the systems as licensing inference to inferred lower probability bounds. According to the present assumption, the precise bound licensed by a respective system, X, relative to a given knowledge base, Δ, and a probability function, P, is as follows, where Δ' ranges over the subsets of Δ such that $\Delta' \vdash_X (\psi|\phi)$:

$$X(\psi|\phi) = \max_{\Delta' \subseteq \Delta} \Big\{ 1 - \sum_{(\psi_i|\phi_i) \in \Delta'} (1 - P(\psi_i|\phi_i)) \Big\}. \tag{4}$$

While the present assumption is 'correct' in the case of System P, it may lead to overestimation when applied to the other three systems. Precisely, we say that inference made by a system counts as an *overestimation*, whenever $X(\psi|\phi) > P(\psi|\phi)$. For the moment, we will proceed as if it is reasonable to evaluate the accuracy of inferred conditionals in the present manner, bearing in mind that any charge of "overestimation" is based on the assumption that it is correct to propogate lower probability bounds in the manner of improbability sums. In the conclusion of the paper, when we consider what to make of our experminental results, we will briefly revisit this assumption.

Beyond attending to cases where a respective system overestimates respective conditional probabilities, our interest is in comparing the accuracy of the bounds licenced by the Systems Z and MinC. Unfortunately, there are no established and uncontroversial measures for scoring the accuracy of lower probability bounds. For this reason, we report the results of a scoring method that has a principled motivation and is pertinent to assessing accuracy, namely, the advantage-compared-to-guessing measure (ACG) [19]:

$$ACG(X(\psi|\phi), P) = \frac{1}{3} - |P(\psi|\phi) - X(\psi|\phi)|. \tag{ACG}$$

The idea behind this measure derives from the fact that the mean difference between two random choices of real values r and s from the unit interval is, provably, $\frac{1}{3}$. This means that the 'strategy' of setting lower probability bounds by randomly choosing numbers in $[0,1]$ is expected to yield an ACG score of zero, on average (assuming that the true probabilities are also selected randomly from $[0, 1]$). Reporting ACG scores, rather than the *linear distance* of inferred bounds from the true probabilities, has heuristic value, since the measure assigns *positive* scores to judgments that are 'better than guessing', and *negative* scores to judgments that are 'worse than guessing'. Given the appearance of X in the calculation of ACG scores, we once again observe that our proposed evaluation assumes that it is correct to propogate lower probability bounds in the manner of improbability sums.

7 Experimental results

The results presented here, regarding systems P, Z, and QC, are similar to those presented in [19]. The results of this paper are novel inasmuch they permit a comparison of the performance of Systems Z and MinC. All tested systems do satisfy certain quality criteria, as noted in Sections 3 and 4, and hence the inferences drawn are sensible with respect to those criteria.

Table 2 presents the number of inferences made by each of the four systems over the course of 5,000 simulations, for each of the listed values of mp (the minimum probability of the conditionals in the knowledge base). Table 2 illustrates that System MinC permits more inferences than System Z, while both systems permit quite a few more inferences than System P, and far fewer inferences than System QC. It may also be observed that the difference between the number of System MinC and System Z inferences decreases with increases in the value of mp. Indeed, if we exclude those inferences that are made by System P, then we see that System MinC licenses about 10% more inferences than System Z, when mp = 0.5. At mp = 0.99, System MinC licenses about 5% more inferences than System Z. At present, we cannot say whether the behavior of System Z and System MinC converge as mp goes to 1.

Every inference licensed by System P is included in each of the other systems. On the other hand, it has been demonstrated that the set of inferences licensed by a minimal c-representation does not generally include those licensed by System Z, and similarly the set of inferences licensed by System Z does not generally include those licensed by a minimal c-representation [10]. Our experiments expand upon this finding, showing that although there are inferences that are licensed by System Z that are not licensed by System MinC, such inferences are rare. Indeed, in addition to licensing more conclusions than System Z, the set of conclusions licensed by System MinC frequently includes the set of conclusions licensed by System Z, as presented in the right most column of Tbl 2.

Example 3. To show that System Z and System MinC are different in general we use an Example from [10]. By applying System Z and System MinC to the knowldege base $\varDelta = \{(a|b), (\overline{a}|c), (b|c), (d|b)\}$, we obtain that $((\overline{c}b \vee c\overline{b}) \wedge \overline{a}\overline{d}) \vdash_Z \ \overline{c}b$ and $((\overline{c}b \vee c\overline{b}) \wedge \overline{a}\overline{d}) \nvdash_{MinC} \overline{c}b$, whereas $c\overline{b}d \vdash_{MinC} \overline{a}$ and $c\overline{b}d \nvdash_Z \overline{a}$.

Table 3 shows that both System Z and MinC are somewhat prone to overestimation, which characterizes the majority of System Z and MinC inferences when the value of mp is high. Table 3 also shows that the inferences made by System MinC tend to be less accurate than those of System Z, as measured by the ACG measure. This fact is partially obscured by the fact that the sets of inferences made by systems Z and MinC

Table 2. Total number of inferred conditionals.

mp	Number of inferred conditionals P	Z	MinC	QC	$\|$Z ∩ MinC$\|$	$\|$Z\MinC$\|$	$\|$MinC\Z$\|$	Cases where Z⊆MinC
0.5	65 777	258 400	278 366	612 815	257 671	729	20 695	4 535
0.6	51 368	232 926	249 612	508 811	232 404	522	17 208	4 627
0.7	39 354	206 108	218 756	423 102	205 783	325	12 973	4 744
0.8	29 899	175 412	184 751	338 832	175 201	211	9 550	4 813
0.9	24 296	133 602	139 197	218 434	133 566	36	5 631	4 965
0.99	20 690	74 368	76 000	92 904	74 368	0	1 632	5 000

Table 3. Aggregate ACG scores and number of overestimations.

mp	Aggregate ACG scores				Overestimations		
	P	Z	MinC	QC	Z	MinC	QC
0.5	9 413.7	31 385.6	33 019.9	33 880.1	94 512	106 767	368 061
0.6	10 556.5	32 564.0	33 852.3	24 691.5	98 270	109 754	329 044
0.7	10 078.2	30 887.3	31 715.8	15 609.1	101 240	110 764	294 497
0.8	8 973.8	27 711.8	28 144.2	7 497.0	100 573	108 520	255 494
0.9	7 888.6	22 808.0	22 823.5	8 987.9	90 520	95 958	174 652
0.99	6 893.0	19 076.3	19 324.0	16 015.8	56 659	58 291	75 195

both include the inferences made by System P, and by the fact that the set of System Z inferences is 'practically' included in the set of System MinC inferences. In order to present a clearer picture, Table 4 presents the average ACG score earned for *individual* inferences made by System P, inferences made by System Z that were not made by System P (Z\P), inferences made by System MinC that were not made by System Z (MinC\Z), and inferences made by System QC that were not made by System MinC (QC\MinC). Here we see that inferences proper to System MinC (MinC\Z) earned positive ACG across all values of mp, as with the inferences proper to System Z (Z\P), and unlike the inferences proper to System QC (QC\MinC).

Finally, since one of our primary concerns was to assess the reasonableness of inheritance inference in the case of exceptional subclasses, we compared the accuracy of the inheritance inferences licensed by System Z (which only involve unexceptional subclasses) with the accuracy of the inheritance inferences licensed by System MinC (which may involve exceptional subclasses). As a means of assessment, we counted an inference to a conditional $(\psi|\phi)$ as inferred by inheritance from a given premise set if and only if (i) $(\psi|\phi)$ was neither a member of the premise set nor inferred from the premise set by System P, and (ii) some conditional $(\psi|\xi)$ was also inferred, where $\phi \models \xi$ and $\xi \not\models \phi$. Information regarding such inferences is recorded in Table 5. Here we see that inheritance inferences make up the majority of the inferences licensed by Systems Z and MinC that are not also licensed by System P, ranging from just over 75% of the total inferences, for mp = 0.5, to just over 50% of the total inferences, for mp = 0.99. It also follows from results of Table 5 that the accuracy of the inheritance inferences licensed by System Z, as measured the average ACG scores per inference, is nearly identical to

Table 4. Mean ACG scores per inference.

mp	P	Z\P	MinC\Z	QC\MinC
0.5	0.143	0.114	0.082	0.003
0.6	0.206	0.121	0.077	−0.035
0.7	0.256	0.125	0.066	−0.079
0.8	0.300	0.129	0.046	−0.134
0.9	0.325	0.136	0.003	−0.175
0.99	0.333	0.227	0.152	−0.196

Table 5. Number of inheritance inferences and their aggregate ACG scores.

mp	Number of inferences			Aggregate ACG scores		
	Z	MinC	QC	Z	MinC	QC
0.5	145 763	163 532	384 009	15 827.1	17 276.2	20 253.2
0.6	134 338	149 252	314 733	16 587.1	17 749.1	15 043.7
0.7	120 407	131 738	259 069	16 103.8	16 867.5	10 200.4
0.8	102 610	111 088	204 322	14 992.7	15 395.3	6 017.1
0.9	72 671	77 896	120 005	11 752.3	11 770.0	6 291.0
0.99	27 021	28 649	35 261	6 758.9	7 006.0	6 315.4

the accuracy of the non-inheritance inferences among Z\P. Similarly, the accuracy of the inheritance inferences licensed by System MinC, and not by System Z, is nearly identical to the accuracy of the non-inheritance inferences among MinC\Z.

In summary, the results of our simulations are as follows (where accuracy claims assume the correctness of probability propogation by improbability sums):

1. System MinC licensed significantly more inferences than System Z, with a decreasing margin proportional to the value of mp.
2. While neither System Z nor System MinC strictly includes the other (as shown in [10]), the set of System Z inferences was a subset of the set of System MinC inferences within a vast majority of our simulations.
3. The accuracy of inferences licensed by System MinC was somewhat less than the accuracy of inferences drawn by System Z. We also observed that the accuracy of System MinC inferences tended to decrease with increasing values of mp (excluding the case where $mp = 0.99$, whose exceptionality is discussed at length in [19, § 2.5]).
4. The accuracy of the inheritance inferences licensed by System Z was nearly identical to that of the other inferences licensed by System Z that were not licensed by System P. Similarly, the accuracy of the inheritance inferences licensed by System MinC was nearly identical to that of the other inferences licensed by System MinC that were not licensed by System Z.

8 Conclusion

Our results show that for practical purposes, System MinC represents a stronger system of inference than System Z. Our results also show that inference by System MinC (and inheritance for exceptional subclasses, as licensed by System MinC) is more risky than inference by System Z. These results accord with existing theoretical analyses of the systems studied here [12,14,10]. Indeed, as measured by the type of monotony that characterize the systems, we see that inferential strength increases as we proceed from System P to Systems Z and MinC, and finally to System QC: *cautious* monotony holds for System P, *rational* monotony holds for Systems Z and MinC, and "full" monotony holds for System QC. As measured by the type of subclass inheritance supported by the systems, we see that inferential strength increases as we proceed from System P to System Z to System MinC and finally to System QC: no inheritance inference is permitted in System P, inheritance inference in the case of unexceptional subclasses is permitted in System Z, defeasible inheritance for exceptional subclasses is permitted in System MinC, and unrestricted inheritance inference is permitted in System QC. Our experimental results show that increasing inferential strength, as described, comes at the risk of decreased accuracy. Assuming the risk associated with such inferential strength is too high in the case of System QC (as argued in [19,22,23]), the question remains of whether inference by System MinC should be favored over inference by System Z.

While inference by System MinC carries greater risk than inference by System Z, the same claim can be made in comparing inference by System Z to inference by System P. In the latter case, the riskiness of inference by System Z appears to be small enough, so that inference by System Z should be preferred to inference by System P (as argued in [19,22,23]), or better: One should perform the inferences licensed by System Z *in addition* to those licensed by System P. Assuming such arguments are cogent in the case of System Z, are similar arguments cogent in the case of System MinC? In other words, should one perform the inferences licensed by System MinC in addition to those licensed by System Z? While we grant that the risks (of overestimation and inaccurate judgment) are greater in the case of System MinC (in comparison to System Z), we also observe that inference by System MinC generally yields positive accuracy scores according to the ACG measure, in the case where probability propogation is determined by improbability sums.

In addition to evaluating the performance of System MinC, we were keen to evaluate the accuracy of inheritance inference in the case of exceptional subclasses. In Section 5, we offered conceptual reasons for rejecting a blanket prohibition of such inferences. Our argument there proceeded from the fact that the class of inheritance inferences to exceptional subclasses is very broad and encompasses many inferences that are generally, and correctly (we maintain), regarded as reasonable. Of course, we do not endorse the wholesale adoption of all inheritance inferences, which would be tantamount to reasoning in accordance with System QC. Our hope is rather that there is some systematic way to move beyond System Z, and a blanket prohibition of inheritance inference in the case of exceptional subclasses. Our motivation for evaluating the performance of System MinC experimentally was to determine whether inference by System MinC might serve as an appropriate means of moving beyond System Z. As things stand (and for the reasons adduced in the preceding paragraphs), we think that inference by System MinC represents a promising option.

Finally, it should be mentioned, once again, that the overestimations and accuracy scores attributed to the studied systems are premised on treating the systems as inferring lower probability bounds in accordance with (4), above. We think that application of (4) is reasonable, since such inferences are valid (i.e., guarenteed to be truth preserving) for System P, and the other systems represent incremental strengthenings of System P. Moreover, the fact that such inferences are invalid in the case of Systems Z, MinC, and QC, is not a decisive objection to the proposed application of (4), since these three systems all license inheritance inference, for which there is never a guarantee that high premise probability is preserved (i.e., $\forall \phi, \psi, \chi, r$: $(r < 1$ and $\phi \nvDash \chi) \Rightarrow (\exists P$: $P(\psi|\phi) = r$ and $P(\psi|\phi \wedge \chi) = 0)$. Nevertheless, while applying (4) yields a plausible means of evaluating the four systems, there are certainly possible alternatives. Exploring such alternatives is an object of present and future research.

Acknowledgment: We thank the anonymous referees for their valuable suggestions that helped us improve the paper. This work was supported by DFG-Grant KI1413/5-1 of Prof. Dr. Gabriele Kern-Isberner, and DFG-Grant SCHU1566/9-1 of Prof. Dr. Gerhard Schurz, both as part of the priority program "New Frameworks of Rationality" (SPP 1516).

References

1. Adams, E.: A note on comparing probabilistic and modal logics of conditionals. Theoria 43, 186–194 (1971)
2. Adams, E.: The Logic of Conditionals: An Application of Probability to Deductive Logic. Synthese Library, Springer (1975)
3. Bacchus, F.: Representing and reasoning with probabilistic knowledge. MIT Press, Cambridge, Massachusetts (1990)
4. Benferhat, S., Cayrol, C., Dubois, D., Lang, J., Prade, H.: Inconsistency management and prioritized syntax-based entailment. In: Proceedings of the International Joint Conferences on Artificial Intelligence. pp. 640–647 (1993)
5. De Finetti, B.: Theory of Probability, vol. 1,2. John Wiley and Sons, New York, USA (1974)
6. Geffner, H., Pearl, J.: Conditional entailment: Bridging two approaches to default reasoning. Artificial Intelligence 53(2–3), 209–244 (1992)
7. Hawthorne, J., Makinson, D.: The quantitative/qualitative watershed for rules of uncertain inference. Studia Logica 86(2), 247–297 (2007)
8. Kern-Isberner, G.: Conditionals in Nonmonotonic Reasoning and Belief Revision – Considering Conditionals as Agents. No. 2087 in LNCS, Springer (2001)
9. Kern-Isberner, G.: A thorough axiomatization of a principle of conditional preservation in belief revision. Annals of Mathematics and Artificial Intelligence 40, 127–164 (2004)
10. Kern-Isberner, G., Eichhorn, C.: Structural inference from conditional knowledge bases. In: Unterhuber, M., Schurz, G. (eds.) Studia Logica Special Issue Logic and Probability: Reasoning in Uncertain Environments, pp. 751–769. No. 102 (4), Springer Science+Business Media, Dordrecht, NL (August 2014)
11. Kuhnmünch, G., Ragni, M.: Can Formal Non-monotonic Systems Properly Describe Human Reasoning? In: Proceedings of the Cognitive Science Conference (COGSCI2014). pp. 1806–1811 (2014)
12. Lehmann, D., Magidor, M.: What does a conditional knowledge base entail? Artificial Intelligence 55(1), 1–60 (1992)
13. Lewis, D.: Counterfactuals. Blackwell, Oxford, UK (1973)

14. Makinson, D.: General patterns in nonmonotonic reasoning. In: Gabbay, D.M., Hogger, C.J., Robinson, J.A. (eds.) Handbook of logic in artificial intelligence and logic programming, vol. 3, pp. 35–110. Oxford University Press, Inc., New York, NY, USA (1994)
15. Müller, C.: Optimierte Wissensdarstellung auf der Basis von Default-Regeln. Masters thesis, FernUniversität Hagen, Hagen, DE (2003), (in German)
16. Pearl, J.: System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning. In: Proceedings of the 3rd conference on Theoretical aspects of reasoning about knowledge. pp. 121–135. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990)
17. Pfeifer, N., Kleiter, G.D.: Coherence and Nonmonotonicity in Human Reasoning. Synthese 146(1–2), 93–109 (2005)
18. Pollock, J.L.: Nomic Probability and the Foundations of Induction. Oxford University Press (1990)
19. Schurz, G., Thorn, P.D.: Reward versus risk in uncertain inference: Theorems and simulations. The Review of Symbolic Logic 5, 574–612 (11 2012)
20. Spohn, W.: The Laws of Belief: Ranking Theory and Its Philosophical Applications. Oxford University Press (2012)
21. Spohn, W.: Ordinal Conditional Functions: A Dynamic Theory of Epistemic States. In: Causation in Decision, Belief Change and Statistics: Proceedings of the Irvine Conference on Probability and Causation. The Western Ontario Series in Philosophy of Science, vol. 42, pp. 105–134. Springer Science+Business Media, Dordrecht, NL (1988)
22. Thorn, P.D., Schurz, G.: Ampliative Inference Under Varied Entropy Levels. In: Beierle, C., Kern-Isberner, G. (eds.) Proceedings of the 4th Workshop on Dynamics of Knowledge and Belief (DBK-2013). pp. 46–60. FernUniversität in Hagen (2013)
23. Thorn, P.D., Schurz, G.: A Utility Based Evaluation of Logico-probabilistic Systems. Studia Logica 102(4), 867–890 (2014)
24. Thorn, P.D., Schurz, G.: Qualitative probabilistic inference under varied entropy levels. Journal of Applied Logic (forthcoming)

Algebraic Semantics for Graded Propositions

Haythem O. Ismail[1,2] and Nourhan Ehab[2]

[1] Cairo University, Department of Engineering Mathematics
[2] German University in Cairo, Department of Computer Science
{haythem.ismail,nourhan.ehab}@guc.edu.eg

Abstract. We present $Log_A\mathbf{G}$, an algebraic language for reasoning about graded propositions. $Log_A\mathbf{G}$ is algebraic in that it is a language of only terms, some of which denote propositions. Both propositions and their grades are taken as individuals in the $Log_A\mathbf{G}$ ontology. Thus, the language includes terms denoting graded propositions, grades of propositions, grading propositions, and graded grading propositions in an arbitrary compositional structure. In this paper, we present the syntax and semantics of $Log_A\mathbf{G}$, defining an infinite sequence of graded logical consequence relations, each corresponding to accepting graded propositions at some nesting depth. We show the utility of $Log_A\mathbf{G}$ in default reasoning, reasoning about information provided by a chain of sources with varying degrees of trust, and representing the dilemma one is in when facing paradoxical liar-like sentences.

1 Introduction

Graded, or weighted, logics have witnessed increased attention and interest over the years, which may be attested by the sheer length of the bibliography of a recent comprehensive survey [1]. Whether the interest is in modelling uncertain beliefs [2, 3, for instance], reasoning with vague predicates [4, 5, for instance], revising a logical theory [6], or jumping to default conclusions [7], weighted logics are always an obvious resort. To demonstrate the variety of phenomena falling under the rubric of weighted logics, we present three problems (two of which are classical) that we shall carefully revisit in Section 4.

The Case of Opus and Tweety. Tweety is a bird and Opus is a penguin. You believe that penguins are birds. In the absence of other information, you would like to jump to the conclusion that Tweety flies and Opus does not. How do you do so gracefully and without succumbing to absurdity?

The Case of Superman. You open the Daily Planet and read a report by Lois Lane claiming that Superman was seen in downtown Metropolis at noon. You happen to have seen Clark Kent at his office at noon, and you have always had a feeling that Superman is Clark Kent. What should you believe about the whereabouts of Superman if you trust your perception very much, you trust Lois Lane's honesty, you only mildly trust the Daily Planet, and you still have your doubts about whether Superman is indeed Clark Kent?

The Case of the Liar. The first sentence of the paragraph titled "The Case of the Liar" in this paper is not true. Having read the previous sentence, should you believe it or not? (cf. [8].)

Weighted logics have something to say about each of the above cases (or, at least, so we claim). In this report, we present the syntax and semantics of a family of logical languages, $Log_A\mathbf{G}$, for reasoning about graded propositions and, in Section 4, we demonstrate how the above cases are treated within $Log_A\mathbf{G}$ theories. While most of the weighted logics we are aware of employ some form of non-classical, possible-worlds, semantics, basically assigning some notion of grade to possible worlds or truth values, $Log_A\mathbf{G}$ is a non-modal logic, with classical notions of worlds and truth values. This is not to say that $Log_A\mathbf{G}$ is a common classical logic—it surely is not—but it is closer in spirit to classical non-monotonic logics in artificial intelligence [9, 7, for example].[3] In such formalisms, as in $Log_A\mathbf{G}$, there is a classical logical consequence relation on top of which we define a non-classical relation which is more restrictive, selecting only a subset of the classical models. We achieve this by taking the algebraic, rather than the modal, route.

$Log_A\mathbf{G}$ is algebraic in the sense that it only contains terms, algebraically constructed from function symbols. No sentences are included in a $Log_A\mathbf{G}$ language; instead, there are terms of a distinguished syntactic type that are taken to denote propositions. $Log_A\mathbf{G}$ is a variant of $Log_A\mathbf{B}$ [10] and $Log_A\mathbf{S}$ [11], which are algebraic languages for reasoning about, respectively, beliefs and temporal phenomena. The inclusion of propositions in the ontology, though non-standard, has been suggested by several authors [12–15, for example]. (See [10] and [15] for a thorough defense of this position.) In the $Log_A\mathbf{G}$ ontology, propositions are structured in a Boolean algebra, giving us, almost for free, all standard truth conditions and standard notions of consequence and validity. In addition, we also admit *grades* as first-class individuals in the ontology. Thus, we combine propositions and grades to construct propositions *about* graded propositions, which, recursively, are themselves gradable. This yields a language that is on one hand quite expressive and, on the other hand, intuitive and very similar in syntax to first-order logic.[4]

2 Log_AG Languages

$Log_A\mathbf{G}$ is a class of many-sorted languages that share a common core of logical symbols and differ in a signature of non-logical symbols. In what follows, we identify a sort σ with the set of symbols of sort σ. A $Log_A\mathbf{G}$ language is a set of terms partitioned into three base syntactic sorts, σ_P, σ_D and σ_I. Intuitively, σ_P

[3] But it is neither second-order like circumscriptive theories [9] nor dependent on special default rules like default logic [7].

[4] While multi-modal logics such as those presented in [16] and [17] may be used to express graded *grading* propositions, the grades themselves are embedded in the modal operators and are not amenable to reasoning and quantification.

is the set of terms denoting propositions, σ_D is the set of terms denoting grades of propositions, and σ_I is the set of terms denoting anything else.

As is customary in many-sorted languages, an alphabet of $Log_A\mathbf{G}$ is made up of a set of syncategorematic punctuation symbols and a set of denoting symbols each from a set $\sigma = \{\sigma_P, \sigma_D, \sigma_I\} \cup \{\tau_1 \longrightarrow \tau_2 | \tau_1 \in \{\sigma_P, \sigma_D, \sigma_I\}$ and $\tau_2 \in \sigma\}$ of syntactic sorts. Intuitively, $\tau_1 \longrightarrow \tau_2$ is the syntactic sort of function symbols that take a single argument of sort σ_P, σ_D, or σ_I and produce a functional term of sort τ_2. Given the restriction of the first argument of function symbols to base sorts, $Log_A\mathbf{G}$ is, in a sense, a first-order language.

A $Log_A\mathbf{G}$ alphabet is a union of four disjoint sets: $\Omega \cup \Xi \cup \Sigma \cup \Lambda$. The set Ω, the *signature* of the language, is a non-empty, countable set of constant and function symbols. Each symbol in the signature has a designated syntactic type from σ. The set $\Xi = \{x_i, d_i, p_i\}_{i \in \mathbb{N}}$ is a countably infinite set of variables, where $x_i \in \sigma_I$, $d_i \in \sigma_D$, and $p_i \in \sigma_P$, for $i \in \mathbb{N}$. Σ is a set of syncategorematic symbols, including the comma, various matching pairs of brackets and parentheses, and the symbol $\forall$. The set Λ is the set of logical symbols of $Log_A\mathbf{G}$, defined as the union of the following sets.

1. $\{\neg\} \subseteq \sigma_P \longrightarrow \sigma_P$
2. $\{\wedge, \vee\} \subseteq \sigma_P \longrightarrow \sigma_P \longrightarrow \sigma_P$
3. $\{\prec, \doteq\} \subseteq \sigma_D \longrightarrow \sigma_D \longrightarrow \sigma_P$
4. $\{\mathbf{G}\} \subseteq \sigma_P \longrightarrow \sigma_D \longrightarrow \sigma_P$

A $Log_A\mathbf{G}$ language with signature Ω is denoted by L_Ω. It is the smallest set of terms formed according to the following rules; as usual, terms involving $\Rightarrow$, $\Leftrightarrow$, and $\exists$ may be introduced as abbreviations in the standard way.

- $\Xi \subset L_\Omega$
- $c \in L_\Omega$, where $c \in \Omega$ is a constant symbol.
- $f(t_1, \ldots, t_n) \in L_\Omega$, where $f \in \Omega$ is of sort $\tau_1 \longrightarrow \ldots \longrightarrow \tau_n \longrightarrow \tau$ $(n > 0)$ and t_i is of sort τ_i.
- $\{\neg t_1, (t_1 \wedge t_2), (t_1 \vee t_2), \forall x(t_1), \mathbf{G}(t_1, t_2), t_3 \prec t_4, t_3 \doteq t_4\} \subset L_\Omega$; where $t_1, t_2 \in \sigma_P$; $t_3, t_4 \in \sigma_D$; and $x \in \Xi$.

The basic ingredient of the $Log_A\mathbf{G}$ semantic apparatus is the notion of a $Log_A\mathbf{G}$ structure.

Definition 1. *A* $Log_A\mathbf{G}$ structure *is a quintuple* $\mathfrak{S} = \langle \mathcal{D}, \mathfrak{A}, \mathfrak{g}, <, \mathfrak{e} \rangle$*, where*

- $\mathcal{D}$*, the domain of discourse, is a set with two disjoint, non-empty, countable subsets* $\mathcal{P}$ *and* $\mathcal{G}$*.*
- $\mathfrak{A} = \langle \mathcal{P}, +, \cdot, -, \perp, \top \rangle$ *is a complete (closed under arbitrary products and sums), non-degenerate* ($\top \neq \perp$) *Boolean algebra [18].*
- $\mathfrak{g} : \mathcal{P} \times \mathcal{G} \longrightarrow \mathcal{P}$.
- $<: \mathcal{G} \times \mathcal{G} \longrightarrow \mathcal{P}$ *satisfies the following properties for every distinct* $g_1, g_2, g_3 \in \mathcal{G}$*:*

  O1. $g_1 < g_2 = -(g_2 < g_1)$.

  O2. $[(g_1 < g_2) \cdot (g_2 < g_3)] + g_1 < g_3 = g_1 < g_3$.

O3. $g_1 < g_1 = \bot$.
O4. $\sum_{g \in \mathcal{G}} g_1 < g = \sum_{g \in \mathcal{G}} g < g_1 = \top$.

– $\mathfrak{e} : \mathcal{G} \times \mathcal{G} \longrightarrow \{\bot, \top\}$, *where, for every* $g_1, g_2 \in \mathcal{G}$, $\mathfrak{e}(g_1, g_2) = \top$, *if* $g_1 = g_2$, *and* $\mathfrak{e}(g_1, g_2) = \bot$, *otherwise.*

Intuitively, the domain $\mathcal{D}$ is partitioned into three cells: (i) a set of propositions $\mathcal{P}$, structured as a Boolean algebra; (ii) a set of grades $\mathcal{G}$, and (iii) a set of individuals $\overline{\mathcal{P} \cup \mathcal{G}}$. These stand in correspondence to the syntactic sorts of $Log_A\mathbf{G}$. In what follows, we let $\mathcal{D}_{\sigma_P} = \mathcal{P}$, $\mathcal{D}_{\sigma_D} = \mathcal{G}$, and $\mathcal{D}_{\sigma_I} = \overline{\mathcal{P} \cup \mathcal{G}}$. $\mathfrak{g}$ is a function which maps a proposition p and a grade g to the proposition that p is a proposition of grade g. By refraining from imposing any constraints on $\mathfrak{g}$ (other than functionality), we are admitting virtually any intuitive interpretation of grading. Properties **O1**–**O4** require propositions in the range of $<$ to give rise to an irreflexive linear order on $\mathcal{G}$ which is serial in both directions. Similarly, the rigid definition of $\mathfrak{e}$ gives rise to the identity relation on $\mathcal{G}$.

Definition 2. *A* valuation $\mathcal{V}$ *of a* $Log_A\mathbf{G}$ *language* L_Ω *is a triple* $\langle \mathfrak{S}, \mathcal{V}_\Omega, \mathcal{V}_\Xi \rangle$, *where*

– $\mathfrak{S} = \langle \mathcal{D}, \mathfrak{A}, \mathfrak{g}, <, \mathfrak{e} \rangle$ *is a* $Log_A\mathbf{G}$ *structure;*
– $\mathcal{V}_\Omega$ *is a function that assigns to each constant of sort* τ *in* Ω *an element of* $\mathcal{D}_\tau$, *and to each function symbol* $f \in \Omega$ *of sort* $\tau_1 \longrightarrow \ldots \longrightarrow \tau_n \longrightarrow \tau$ *an* n*-adic function* $\mathcal{V}_\Omega(f) : \bigtimes_{i=1}^{n} \mathcal{D}_{\tau_i} \longrightarrow \mathcal{D}_\tau$; *and*
– $\mathcal{V}_\Xi : \Xi \longrightarrow \mathcal{D}$ *is a variable assignment, where, for every* $i \in \mathbb{N}$, $v_\Xi(p_i) \in \mathcal{D}_{\sigma_P}$, $v_\Xi(d_i) \in \mathcal{D}_{\sigma_D}$, *and* $v_\Xi(x_i) \in \mathcal{D}_{\sigma_I}$.

In what follows, for a valuation $\mathcal{V} = \langle \mathfrak{S}, \mathcal{V}_\Omega, \mathcal{V}_\Xi \rangle$ with $x \in \Xi$ of sort τ and $a \in \mathcal{D}_\tau$, $\mathcal{V}[a/x] = \langle \mathfrak{S}, \mathcal{V}_\Omega, \mathcal{V}_\Xi[a/x] \rangle$, where $\mathcal{V}_\Xi[a/x](x) = a$, and $\mathcal{V}_\Xi[a/x](y) = \mathcal{V}_\Xi(y)$ for every $y \neq x$.

Definition 3. *Let* L_Ω *be a* $Log_A\mathbf{G}$ *language and let* $\mathcal{V}$ *be a valuation of* L_Ω. *An* interpretation *of the terms of* L_Ω *is given by a function* $[\![\cdot]\!]^\mathcal{V}$:

– $[\![x]\!]^\mathcal{V} = \mathcal{V}_\Xi(x)$, *for* $x \in \Xi$
– $[\![c]\!]^\mathcal{V} = \mathcal{V}_\Omega(c)$, *for a constant* $c \in \Omega$
– $[\![f(t_1, \ldots, t_n)]\!]^\mathcal{V} = \mathcal{V}_\Omega(f)([\![t_1]\!]^\mathcal{V}, \ldots, [\![t_n]\!]^\mathcal{V})$, *for an* n*-adic* $(n \geq 1)$ *function symbol* $f \in \Omega$
– $[\![(t_1 \wedge t_2)]\!]^\mathcal{V} = [\![t_1]\!]^\mathcal{V} \cdot [\![t_2]\!]^\mathcal{V}$
– $[\![(t_1 \vee t_2)]\!]^\mathcal{V} = [\![t_1]\!]^\mathcal{V} + [\![t_2]\!]^\mathcal{V}$
– $[\![\neg t]\!]^\mathcal{V} = -[\![t]\!]^\mathcal{V}$
– $[\![\forall x(t)]\!]^\mathcal{V} = \prod_{a \in \mathcal{D}_\tau} [\![t]\!]^{\mathcal{V}[a/x]}$
– $[\![\mathbf{G}(t_1, t_2)]\!]^\mathcal{V} = \mathfrak{g}([\![t_1]\!]^\mathcal{V}, [\![t_2]\!]^\mathcal{V})$
– $[\![t_1 \prec t_2]\!]^\mathcal{V} = [\![t_1]\!]^\mathcal{V} < [\![t_2]\!]^\mathcal{V}$
– $[\![t_1 \doteq t_2]\!]^\mathcal{V} = \mathfrak{e}([\![t_1]\!]^\mathcal{V}, [\![t_2]\!]^\mathcal{V})$

In $Log_A\mathbf{G}$, logical consequence is defined in pure algebraic terms without alluding to the notion of truth. This is achieved using the natural partial order $\leq$ associated with $\mathfrak{A}$ [18], where, for $p_1, p_2 \in \mathcal{P}$, $p_1 \leq p_2 =_{\text{def}} p_1 \cdot p_2 = p_1$.

Definition 4. *Let L_Ω be a $Log_A\mathbf{G}$ language. For every $\phi \in \sigma_P$ and $\Gamma \subseteq \sigma_P$, ϕ is a logical consequence of Γ, denoted $\Gamma \models \phi$, if, for every L_Ω valuation $\mathcal{V}$, $\prod_{\gamma\in\Gamma} [\![\gamma]\!]^{\mathcal{V}} \leq [\![\phi]\!]^{\mathcal{V}}$.*

In [10], it is shown that $\models$ has the distinctive properties of classical Tarskian logical consequence and that it satisfies a counterpart of the deduction theorem.

3 Graded Filters

In what follows, for every $p \in \mathcal{P}$ and $g \in \mathcal{G}$, we say that $\mathfrak{g}(p, g)$ *grades* p and that $\mathfrak{g}(p, g)$ is a *grading proposition.* Moreover, if $\mathfrak{g}(p, g) \in \mathcal{Q} \subseteq \mathcal{P}$, we say that p is *graded* in $\mathcal{Q}$. We define the set of *p graders in $\mathcal{Q}$* to be the set $G(p, \mathcal{Q}) = \{q | q \in \mathcal{Q}$ and q grades $p\}$. Throughout, we assume a $Log_A\mathbf{G}$ structure $\mathfrak{S} = \langle \mathcal{D}, \mathfrak{A}, \mathfrak{g}, <, \mathfrak{e}\rangle$.

According to Definition 4, the set of logical consequences of a set Γ of σ_P-terms corresponds to the filter $F([\![\Gamma]\!])$ generated by the set $[\![\Gamma]\!]$ of denotations of members of Γ [18]. In order to accommodate a richer, non-classical set of consequences which includes some *acceptable* propositions graded in $[\![\Gamma]\!]$, we need a more liberal notion of *graded filters.*

3.1 Embedding and Chains

In order to develop the notion of a graded filter, we need to sharpen our intuitions about the nesting structure of propositions graded in a given set.

Definition 5. *A proposition $p \in \mathcal{P}$ is embedded in $\mathcal{Q} \subseteq \mathcal{P}$ if (i) $p \in \mathcal{Q}$ or (ii) for some $g \in \mathcal{G}$, $\mathfrak{g}(p, g)$ is embedded in $\mathcal{Q}$. Henceforth, let $E(\mathcal{Q}) = \{p | p$ is embedded in $\mathcal{Q}\}$.*

Definition 6. *For $\mathcal{Q} \subseteq \mathcal{P}$, let $\delta_{\mathcal{Q}} : E(\mathcal{Q}) \longrightarrow \mathbb{N}$, where*

1. *if $p \in \mathcal{Q}$, then $\delta_{\mathcal{Q}}(p) = 0$; and*
2. *if $p \notin \mathcal{Q}$, then $\delta_{\mathcal{Q}}(p) = e + 1$, where $e = \min_{q \in G(p, E(\mathcal{Q}))}\{\delta_{\mathcal{Q}}(q)\}$.*

$\delta_{\mathcal{Q}}(p)$ is referred to as the degree of embedding of p in $\mathcal{Q}$.

In the sequel, we let $E^n(\mathcal{Q}) = \{p \in E(\mathcal{Q}) \mid \delta_{\mathcal{Q}}(p) \leq n\}$, for every $n \in \mathbb{N}$.

Definition 7. *A grading chain of $p \in \mathcal{P}$ is a finite sequence $\langle q_0, q_1, \ldots, q_n\rangle$ of grading propositions such that q_n grades p and q_i grades q_{i+1}, for $0 \leq i < n$. $\langle q_0, q_1, \ldots, q_n\rangle$ is a grading chain if it is a grading chain of some $p \in \mathcal{P}$.*

A grading chain $C_2 = \langle q_0, q_1, \ldots, q_n \rangle$ *extends* a grading chain C_1 if C_1 is a grading chain of q_0. The grading chain $C_1 \odot C_2$ is said to be an extension of C_1 (where $\odot$ denotes sequence concatenation).

We say that C is a grading chain in $\mathcal{Q}$, if every proposition in C is in $\mathcal{Q}$. Given that we impose no special restrictions on the function $\mathfrak{g}$ and the proposition algebra, grading chains in a set $\mathcal{Q}$ may, in general, be quite counter-intuitive. Hence, we need to introduce some especially interesting sets of propositions.

Definition 8. *Let $\mathcal{Q} \subseteq \mathcal{P}$.*

1. *A grading chain is well-founded if all its extensions are well-founded. $\mathcal{Q}$ is well-founded if every grading chain in $\mathcal{Q}$ is well-founded.*
2. *A grading chain $\langle q_0, q_1, \ldots, q_n \rangle$ is acyclic if, for every $0 \leq i, j \leq n$, $q_i = q_j$ only if $i = j$. $\mathcal{Q}$ is acyclic if every grading chain in $\mathcal{Q}$ is acyclic.*
3. *$\mathcal{Q}$ is depth-bounded if there is some $d \in \mathbb{N}$ such that every grading chain in $\mathcal{Q}$ has at most d distinct grading propositions.*
4. *$\mathcal{Q}$ is fan-out-bounded if there is some $f_{\text{out}} \in \mathbb{N}$ such that every grading proposition in $\mathcal{Q}$ grades at most f_{out} propositions.*
5. *$\mathcal{Q}$ is fan-in bounded if there is some $f_{\text{in}} \in \mathbb{N}$ where $|G(p, \mathcal{Q})| \leq f_{\text{in}}$, for every $p \in \mathcal{Q}$.*
6. *$\mathcal{Q}$ is non-explosive if for every $\mathcal{R} \subseteq \mathcal{Q}$, if $\mathcal{R}$ has finitely-many grading propositions, then so does $F(\mathcal{R})$.*

Given the above notions, if $p \in E(\mathcal{Q})$, then a grading chain C of p in $\mathcal{Q}$ is a *longest grading chain* of p in $\mathcal{Q}$ if

1. C is acyclic; and
2. if C extends a grading chain C', then $C' \odot C$ is not acyclic.

Proposition 1. [5] *Let $\mathcal{Q} \subseteq \mathcal{P}$.*

1. *If $\mathcal{P}$ is depth-bounded, then it is not acyclic.*
2. *If $\mathcal{Q}$ is depth-bounded and acyclic, then $\mathcal{Q}$ is well-founded.*
3. *If $E(\mathcal{Q})$ is depth-bounded, then there is some $n \in \mathbb{N}$ such that $\delta_{\mathcal{Q}}(p) \leq n$, for every $p \in E(\mathcal{Q})$.*
4. *If $\mathcal{P}$ is fan-out-bounded and $\mathcal{Q}$ is non-explosive with finitely-many grading propositions, then $E^n(F(\mathcal{Q}))$ has finitely-many grading propositions, for every $n \in \mathbb{N}$. Further, if $E(F(\mathcal{Q}))$ is depth-bounded, then it has finitely-many grading propositions.*
5. *If $\mathcal{Q}$ is depth-bounded, then every graded $p \in \mathcal{Q}$ has a longest grading chain in $\mathcal{Q}$. Further, if $\mathcal{Q}$ is fan-in-bounded, then every graded $p \in \mathcal{Q}$ has finitely-many longest grading chains in $\mathcal{Q}$.*

Note that, for the existence of longest grading chains, it suffices to have a single, bounded grading chain of p.

[5] Proofs of observations and propositions are omitted for space limitations. A longer version of the paper includes all the proofs.

3.2 Telescoping

The key to defining graded filters is the intuition that the set of consequences of a proposition set $\mathcal{Q}$ may be further enriched by *telescoping* $\mathcal{Q}$ and accepting some of the propositions embedded therein. For this, we need to define (i) the process of telescoping, which is a step-wise process that considers propositions at increasing degrees of embedding, and (ii) a criterion for accepting embedded propositions which, as should be expected, depends on the grades of said propositions.

Definition 9. *Let $\mathfrak{S}$ be a Log_A**G** structure with a depth- and fan-out-bounded $\mathcal{P}$. A telescoping structure for $\mathfrak{S}$ is a quadruple $\mathfrak{T} = \langle \mathcal{T}, \mathfrak{O}, \otimes, \oplus \rangle$, where*

- *$\mathcal{T} \subseteq \mathcal{P}$;*
- *$\mathfrak{O}$ is an ultrafilter of the subalgebra induced by Range($<$) (see [18]);*
- *$\otimes : \bigcup_{i=1}^{\infty} \mathcal{G}^i \longrightarrow \mathcal{G}$; and*
- *$\oplus : \bigcup_{i=1}^{\infty} \mathcal{G}^i \longrightarrow \mathcal{G}$ is commutative in the sense that $\oplus(t) = \oplus(\pi(t))$, where $\pi(t)$ is any permutation of the tuple t.*

Definition 10. *Let $\otimes : \bigcup_{i=1}^{\infty} \mathcal{G}^i \longrightarrow \mathcal{G}$, and let $C = \langle q_0, q_1, \ldots, q_n \rangle$ be a grading chain of $p \in \mathcal{P}$. The fused $\otimes$-grade of p with respect to C is the grade $\mathfrak{f}_{\otimes}(p, C) = \otimes(\langle g_0, \ldots, g_n \rangle)$, where $q_i = \mathfrak{g}(q_{i+1}, g_i)$, for $0 \leq i < n$, and $q_n = \mathfrak{g}(p, g_n)$.*

Definition 11. *Let $\mathfrak{T}$ be a telescoping structure. If $p \in \mathcal{Q}$, for a fan-in-bounded $\mathcal{Q} \subset \mathcal{P}$, then the $\mathfrak{T}$-fused grade of p in $\mathcal{Q}$ is defined as*

$$\mathfrak{f}_{\mathfrak{T}}(p, \mathcal{Q}) = \bigoplus \langle \mathfrak{f}_{\otimes}(p, C_k) \rangle_{k=1}^{n}$$

where $\langle C_k \rangle_{k=1}^{n}$ is a permutation of the set of longest grading chains of p in $\mathcal{Q}$.[6]

Recasting the familiar notion of a *kernel* of a belief base [19] into the context of Log_A**G** structures, we say that a kernel of $\mathcal{Q} \subseteq \mathcal{P}$ is a subset-minimal $\mathcal{X} \subseteq \mathcal{Q}$ such that $F(\mathcal{X})$ is improper ($\neq \mathcal{P}$). Let $\mathcal{Q}^{\perp\!\!\perp}\perp$ be the set of $\mathcal{Q}$ kernels.

Definition 12. *For a telescoping structure $\mathfrak{T} = \langle \mathcal{T}, \mathfrak{O}, \otimes, \oplus \rangle$ and a fan-in-bounded $\mathcal{Q} \subseteq \mathcal{P}$, if $\mathcal{X} \subseteq \mathcal{Q}$, then $p \in \mathcal{X}$* survives $\mathcal{X}$ in $\mathfrak{T}$ *if*

1. *$p \in \mathcal{T}$; or*
2. *$G(p, \mathcal{Q}) \neq \varnothing$ and there is some $q \in \mathcal{X}$, with $G(q, \mathcal{Q}) \neq \varnothing$, such that $(\mathfrak{f}_{\mathfrak{T}}(q, \mathcal{Q}) < \mathfrak{f}_{\mathfrak{T}}(p, \mathcal{Q})) \in \mathfrak{O}$.*

The set of kernel survivors of $\mathcal{Q}$ in $\mathfrak{T}$ is the set

$$\kappa(\mathcal{Q}, \mathfrak{T}) = \{p \in \mathcal{Q} | \text{ if } p \in \mathcal{X} \in \mathcal{Q}^{\perp\!\!\perp}\perp \text{ then } p \text{ survives } \mathcal{X} \text{ given } \mathfrak{T}\}.$$

Observation 1 *If $F(\mathcal{T})$ is proper, then $F(\kappa(\mathcal{Q}, \mathfrak{T}))$ is proper.*

Definition 13. *Let $\mathcal{Q}, \mathcal{T} \subseteq \mathcal{P}$. We say that p is supported in $\mathcal{Q}$ given $\mathcal{T}$ if*

[6] Note that $\mathfrak{T}$-fusion is well-defined given the fan-in-boundedness of $\mathcal{Q}$ and the final clause of Proposition 1.

1. $p \in F(\mathcal{T})$; or
2. there is a grading chain $\langle q_0, q_1, \ldots, q_n\rangle$ of p in $\mathcal{Q}$ with $q_0 \in F(\mathcal{R})$ where every member of $\mathcal{R}$ is supported in $\mathcal{Q}$.

The set of propositions supported in $\mathcal{Q}$ given $\mathcal{T}$ is denoted by $\varsigma(\mathcal{Q}, \mathcal{T})$.

The following simple observation will prove useful later.

Observation 2 *$\varsigma(\mathcal{Q}, \mathcal{T}) = F(\mathcal{T}) \cup G$, for some set G of propositions graded in $\mathcal{Q}$.*

Definition 14. *Let $\mathfrak{T}$ be a telescoping structure for $\mathfrak{S}$. If $\mathcal{Q} \subset \mathcal{P}$ such that $E^1(F(\mathcal{Q}))$ is fan-in-bounded, then the $\mathfrak{T}$-induced telescoping of $\mathcal{Q}$ is given by $\tau_{\mathfrak{T}}(\mathcal{Q}) = \varsigma(\kappa(E^1(F(\mathcal{Q})), \mathfrak{T}), \mathcal{T})$*

Proposition 2. *For a telescoping structure $\mathfrak{T}$, $\tau_{\mathfrak{T}}$ is a function from fan-in-bounded sets in $2^{\mathcal{P}}$ to sets in $2^{\mathcal{P}}$.*

It may be shown that if $\mathcal{Q}$ is non-explosive with finitely-many grading propositions, then $\tau_{\mathfrak{T}}(\mathcal{Q})$ is defined, for every telescoping structure $\mathfrak{T}$. On the other hand, if $F(\mathcal{Q})$ is improper, then $\tau_{\mathfrak{T}}(\mathcal{Q})$ is undefined. In what follows, provided that the right-hand side is defined, let

$$\tau^n_{\mathfrak{T}}(\mathcal{Q}) = \begin{cases} \mathcal{Q} & \text{if } n = 0 \\ \tau_{\mathfrak{T}}(\tau^{n-1}_{\mathfrak{T}}(\mathcal{Q})) & \text{otherwise} \end{cases}$$

Definition 15. *Let $\mathfrak{T}$ be a telescoping structure. We refer to $F(\tau^n_{\mathfrak{T}}(\mathcal{T}))$ as a degree $n(\in \mathbb{N})$ graded filter of $\mathfrak{T}$, denoted $\mathfrak{F}^n(\mathfrak{T})$.*

Unfortunately, even with a finite and fan-in-bounded $\mathcal{T}$, the existence of a fixed-point for graded filters is not secured. (Check Example 3 in Section 4.) We can only prove a weaker property for a special class of telescoping structures.

Theorem 1. *Let $\mathfrak{T}$ be a telescoping structure where $\mathcal{T}$ is finite. There is some $n \in \mathbb{N}$ such that if $\mathfrak{F}^i(\mathfrak{T})$ is defined and $\mathfrak{F}^i(\mathfrak{T}) \cap Range(\mathfrak{g}) = \tau^i(\mathfrak{T}) \cap Range(\mathfrak{g})$ for every $i \leq n$, then for every $j \in \mathbb{N}$, there is some $k \leq n$ such that $\mathfrak{F}^{n+j}(\mathfrak{T}) = \mathfrak{F}^k(\mathfrak{T})$.*

A fixed-point is guaranteed if, under the same conditions in Theorem 1, we happen to stumble upon a *maximal* graded filter in the following sense.

Corollary 1. *If, in Theorem 1, $\mathfrak{F}^n(\mathfrak{T}) = F(E(\mathcal{T}))$ for some $n < 2^{|E(\mathcal{T})|} + 1$, then $\mathfrak{F}^{n+k}(\mathfrak{T}) = \mathfrak{F}^n(\mathfrak{T})$, for every $k \in \mathbb{N}$.*

Telescoping can never generate an inconsistent theory if the top theory is consistent.

Theorem 2. *If $\mathfrak{T}$ is a telescoping structure where $F(\mathcal{T})$ is proper, then, if defined, $\mathfrak{F}^n(\mathfrak{T})$ is proper, for every $n \in \mathbb{N}$.*

4 $\boldsymbol{Log_A}$G Theories

Given the definition of a $Log_A\mathbf{G}$ structure, we impose some reasonable constraints on which sets of $Log_A\mathbf{G}$ terms qualify as $Log_A\mathbf{G}$ *theories*. A $Log_A\mathbf{G}$ theory is a finite set $\mathbb{T} \subseteq \sigma_P$ such that $\mathbb{E} \cup \mathbb{O} \subseteq \mathbb{T}$, where

- $\mathbb{E}$ is the smallest set containing the following terms:
  1. $\forall d[d \doteq d]$
  2. $\forall d_1, d_2[d_1 \doteq d_2 \Rightarrow d_2 \doteq d_1]$
  3. $\forall d_1, d_2, d_3[(d_1 \doteq d_2 \wedge d_2 \doteq d_3) \Rightarrow d_1 \doteq d_3]$
  4. $\forall p, d_1, d_2[(d_1 \doteq d_2 \wedge \mathbf{G}(p, d_1)) \Rightarrow \mathbf{G}(p, d_2)]$

  and
- $\mathbb{O}$ is the smallest set containing the following terms:
  1. $\forall d_1, d_2[\neg(d_1 \prec d_2) \Leftrightarrow (d_2 \prec d_1 \vee d_2 \doteq d_1)]$
  2. $\forall d_1, d_2, d_3[(d_1 \prec d_2 \wedge d_2 \prec d_3) \Rightarrow d_1 \prec d_3]$

Given a $Log_A\mathbf{G}$ theory $\mathbb{T}$ and a valuation $\mathcal{V} = \langle \mathfrak{S}, \mathcal{V}_\Omega, \mathcal{V}_\Xi \rangle$, let $\mathcal{V}(\mathbb{T}) = \{[\![\phi]\!]^\mathcal{V} \mid \phi \in \mathbb{T}\})$. Further, for a $Log_A\mathbf{G}$ structure $\mathfrak{S}$, an $\mathfrak{S}$ grading canon is a triple $\mathcal{C} = \langle \otimes, \oplus, n \rangle$ where $n \in \mathbb{N}$ and $\otimes$ and $\oplus$ are as indicated in Definition 9.

Definition 16. *Let $\mathbb{T}$ be a $Log_A\mathbf{G}$ theory and $\mathcal{V} = \langle \mathfrak{S}, \mathcal{V}_\Omega, \mathcal{V}_\Xi \rangle$ a valuation, where $\mathfrak{S}$ has a set $\mathcal{P}$ which is depth- and fan-out-bounded, for some $Log_A\mathbf{G}$ language L_Ω. For every $\phi \in \sigma_P$ and $\mathfrak{S}$ grading canon $\mathcal{C} = \langle \otimes, \oplus, n \rangle$, ϕ is a graded consequence of $\mathbb{T}$ with respect to $\mathcal{C}$, denoted $\mathbb{T} \mathrel{|\!\!\approx}^{\mathcal{C}} \phi$, if $\mathfrak{F}^n(\mathfrak{T})$ is defined and $[\![\phi]\!]^\mathcal{V} \in \mathfrak{F}^n(\mathfrak{T})$, for every telescoping structure $\mathfrak{T} = \langle \mathcal{V}(\mathbb{T}), \mathfrak{O}, \otimes, \oplus \rangle$ for $\mathfrak{S}$, where $\mathfrak{O}$ extends $F(\mathcal{V}(\mathbb{T}) \cap Range(<))$.*[7]

It should be clear that $\mathrel{|\!\!\approx}^{\mathcal{C}}$, where $\mathcal{C} = \langle \otimes, \oplus, n \rangle$, reduces to $\models$ if $n = 0$ or if $F(E(\mathcal{V}(\mathbb{T})))$ does not contain any grading propositions. Further, for $n > 0$, no ϕ is a graded consequence of $\mathbb{T}$ with respect to $\mathcal{C}$ if $F(\mathcal{V}(\mathbb{T}))$ is not proper. In what follows, let $\mathbb{T}^\mathcal{C} = \{\phi \mid \mathbb{T} \mathrel{|\!\!\approx}^{\mathcal{C}} \phi\}$. When we are considering a set of canons which only differ in the value of n, we write $\mathbb{T}^n$ instead of $\mathbb{T}^\mathcal{C}$.

Unlike $\models$, $\mathrel{|\!\!\approx}^{\mathcal{C}}$ is, in general, non-monotonic. (In the sequel, we interpret grades by the rational numbers, with their natural order remaining implicit.)

Example 1 (Opus and Tweety). We can represent the case of Opus and Tweety from Section 1 using a $Log_A\mathbf{G}$ theory $\mathbb{T}_{\mathrm{OT1}} = \mathbb{E} \cup \mathbb{O} \cup \Gamma_{\mathrm{OT1}}$, where Γ_{OT1} is made up of the following terms.

1. $\forall x[Bird(x) \Rightarrow \mathbf{G}(Flies(x), 5)]$
2. $\forall x[Penguin(x) \Rightarrow \mathbf{G}(\neg Flies(x), 10)]$
3. $\forall x[Penguin(x) \Rightarrow Bird(x)]$
4. $Penguin(Opus)$
5. $Bird(Tweety)$

Figure 1 displays the relevant graded consequences of $\mathbb{T}_{\mathrm{OT1}}$ with respect to a series of canons, with $0 \leq n \leq 2$. Upon telescoping to $n = 1$, we believe

[7] An ultrafilter U extends a filter F, if $F \subseteq U$.

$n = 0$	0.1. $\mathbb{T}_{\mathrm{OT1}}$ 0.2. $Bird(Opus)$ 0.3. $\mathbf{G}(Flies(Tweety), 5)$ 0.4. $\mathbf{G}(Flies(Opus), 5)$ 0.5. $\mathbf{G}(\neg Flies(Opus), 10)$
$n = 1$	1.1. $\mathbb{T}^0_{\mathrm{OT1}}$ 1.2. $Flies(Tweety)$ 1.3. $\neg Flies(Opus)$

Fig. 1. Graded consequences of the $Log_A\mathbf{G}$ theory $\mathbb{T}_{\mathrm{OT1}}$ from Example 1

that Tweety flies and Opus does not fly. The embedded proposition that Opus flies does not survive telescoping since we trust that Opus does not fly, being a penguin, more than we trust that it flies, being a bird. $\mathbb{T}^1_{\mathrm{OT1}}$ is a fixed point. Now, consider the theory $\mathbb{T}_{\mathrm{OT2}} = \mathbb{E} \cup \mathbb{O} \cup \Gamma_{\mathrm{OT2}}$, where Γ_{OT2} is similar to Γ_{OT1}, but with propositions (1) and (2) replaced by "$\mathbf{G}(\forall x[Bird(x) \Rightarrow Flies(x), 5)$" and "$\mathbf{G}(\forall x[Penguin(x) \Rightarrow \neg Flies(x), 10)$", respectively. Thus, we trade the "*de re*" representation of $\mathbb{T}_{\mathrm{OT1}}$ for the "*de dicto*" representation in $\mathbb{T}_{\mathrm{OT2}}$. This change results in a change in the fixed point that we reach. In $\mathbb{T}^1_{\mathrm{OT2}}$, as in $\mathbb{T}^1_{\mathrm{OT1}}$, we end up believing that Opus does not fly. Unlike $\mathbb{T}^1_{\mathrm{OT1}}$ however, we give up our belief in the proposition that birds fly and, hence, cannot conclude that Tweety flies. □

Example 1 showcases the use of graded propositions in default reasoning. The following example illustrates the utility of nested grading.

Example 2 (Superman). Recalling the case of Superman from Section 1, we can describe the situation using $Log_A\mathbf{G}$ in at least two theories. Consider the theory $\mathbb{T}_{\mathrm{SM1}} = \mathbb{E} \cup \mathbb{O} \cup \Gamma_{\mathrm{SM1}}$, where Γ_{SM1} is made up of the following terms.

1. $\forall p[Source(p, LL) \Rightarrow \mathbf{G}(p, 11)]$
2. $\forall p[Source(p, DP) \Rightarrow \mathbf{G}(p, 4)]$
3. $\forall p[Perceive(p) \Rightarrow \mathbf{G}(p, 15)]$
4. $\forall l, t[\mathbf{G}(At(KC, l, t) \Leftrightarrow At(SM, l, t), 10.5)]$
5. $\forall l_1, l_2, t, x[(Disjoint(l_1, l_2) \wedge At(x, l_1, t)) \Rightarrow \neg At(x, l_2, t)]$
6. $Perceive(Source(Source(At(SM, DT, 12:00), LL), DP))$
7. $Perceive(At(KC, Office, 12:00))$
8. $Disjoint(Office, DT)$

Figure 2 displays relevant members of $\mathbb{T}^n_{\mathrm{SM1}}$ with respect to a series of canons, with $\otimes =$ mean and $\oplus =$ max, and with $0 \leq n \leq 3$. Note that, for $1 \leq n \leq 2$, we trust the proposition that Superman was at the office at noon (1.6). However, upon telescoping to $n = 3$, we lose our trust in said proposition, since we trust what Lois Lane says more than we trust our belief in the identity of Superman and Clark Kent (1.3).

Alternatively, consider the theory $\mathbb{T}_{\mathrm{SM2}} = \mathbb{E} \cup \mathbb{O} \cup \Gamma_{\mathrm{SM2}}$, where Γ_{SM2} is made up of the following terms.

1. $\mathbf{G}(\mathbf{G}(\mathbf{G}(At(SM, DT, 12:00), 11), 4), 15)$

$n = 0$	0.1. $\mathbb{T}_{\mathrm{SM1}}$ 0.2. $\mathbf{G}(Source(Source(At(SM, MDT, 12:00), LL), DP), 15)$ 0.3. $\mathbf{G}(At(KC, Office, 12:00) \Leftrightarrow At(SM, Office, 12:00), 10.5)$ 0.4. $\mathbf{G}(At(KC, Office, 12:00), 15)$
$n = 1$	1.1. $\mathbb{T}^0_{\mathrm{SM1}}$ 1.2. $Source(Source(At(SM, DT, 12:00), LL), DP)$ 1.3. $At(KC, Office, 12:00) \Leftrightarrow At(SM, Office, 12:00)$ 1.4. $At(KC, Office, 12:00)$ 1.5. $\mathbf{G}(Source(At(SM, DT, 12:00), LL), 4)$ 1.6. $At(SM, Office, 12:00)$
$n = 2$	2.1. $\mathbb{T}^1_{\mathrm{SM1}}$ 2.2. $Source(At(SM, DT, 12:00), LL)$ 2.3. $\mathbf{G}(At(SM, DT, 12:00), 11)$
$n = 3$	3.1. Everything at $n = 2$ except 1.3, 1.6, and anything they support 3.2. $At(SM, DT, 12:00)$

Fig. 2. Graded consequences of the $Log_A\mathbf{G}$ theory $\mathbb{T}_{\mathrm{SM1}}$ from Example 2

2. $\mathbf{G}(At(KC, Office, 12:00), 15)$
3. $\forall l, t[\mathbf{G}(At(KC, l, t) \Leftrightarrow At(SM, l, t), 10.5)]$
4. $\forall l_1, l_2, t, x[(Disjoint(l_1, l_2) \wedge At(x, l_1, t)) \Rightarrow \neg At(x, l_2, t)]$
5. $Disjoint(Office, DT)$

Figure 3 displays the different consequences with respect to the same canons employed with $\mathbb{T}_{\mathrm{SM1}}$. In this case, we get a different fixed point; we end up (at $n = 2$) believing that Superman was at the office at noon, contrary to what has been reported by Lois Lane in the Daily Planet. The reason is that, due to the nesting of grading propositions, the fused grade of "$AT(SM, DT, 12:00)$" is now only 10 (which is less than 10.5, the grade of (1.4)), being pulled down by the low grade attributed to the Daily Planet. □

$n = 0$	0.1. $\mathbb{T}_{\mathrm{SM2}}$ 0.2. $\mathbf{G}(At(KC, Office, 12:00) \Leftrightarrow At(SM, Office, 12:00), 10.5)$
$n = 1$	1.1. $\mathbb{T}^0_{\mathrm{SM2}}$ 1.2. $\mathbf{G}(\mathbf{G}(At(SM, DT, 12:00), 11), 4)$ 1.3. $At(KC, Office, 12:00)$ 1.4. $At(KC, Office, 12:00) \Leftrightarrow At(SM, Office, 12:00)$ 1.5. $At(SM, Office, 12:00)$ 1.6. $\neg At(SM, DT, 12:00)$
$n = 2$	2.1. $\mathbb{T}^1_{\mathrm{SM2}}$ 2.2. $\mathbf{G}(At(SM, DT, 12:00), 11)$
$n = 3$	3.1. $\mathbb{T}^2_{\mathrm{SM2}}$

Fig. 3. Graded consequences of the $Log_A\mathbf{G}$ theory $\mathbb{T}_{\mathrm{SM2}}$ from Example 2

$n = 0$	0.1. $\mathbb{T}_{\mathrm{L}}$
$n = 1$	1.1. $\mathbb{T}^0_{\mathrm{L}}$
	1.2. ϕ
	1.3. $\mathbf{G}(\neg\phi, 5)$
$n = 2$	2.1. $\mathbb{T}^0_{\mathrm{L}}$

Fig. 4. Graded consequences of the $Log_A\mathbf{G}$ theory $\mathbb{T}_{\mathrm{L}}$ from Example 3

Finally, we revisit the pathological case of the liar.

Example 3 (The Liar). Consider the theory $\mathbb{T}_{\mathrm{L}} = \mathbb{E} \cup \mathbb{O} \cup \{\mathbf{G}(\phi, 5), \phi \Leftrightarrow \mathbf{G}(\neg\phi, 5)\}$. This is the closest we can get, within $Log_A\mathbf{G}$, to the case of the liar from Section 1; given the non-degeneracy of the Boolean algebra, we can never have the situation where $[\![\phi]\!]^{\mathcal{V}} = -[\![\phi]\!]^{\mathcal{V}}$ (cf. [10]). Figure 4 shows what happens as n increases, given any grading canon. In such a problematic situation, we never reach a fixed point, indefinitely iterating through $\mathbb{T}^0_{\mathrm{L}}$ and $\mathbb{T}^1_{\mathrm{L}}$. But this is just as well, for it fairly captures the dilemma one is in when encountering liar-like sentences. While this is similar in spirit, but not identical, to the treatment of the liar paradox within the revision theory of truth (RTT) [20], RTT tackles the liar paradox head-on, not via the more tame version expressible in $Log_A\mathbf{G}$. (Also see [21] for a treatment of the liar paradox within a fuzzy logical framework.) □

5 Conclusion

Notwithstanding the abundance of weighted logics in the literature, it is our conviction that $Log_A\mathbf{G}$ provides an interesting alternative. While it has a non-classical semantics, $Log_A\mathbf{G}$ is arguably intuitive, expressive, and quite similar in syntax to first-order logic. We hope to have demonstrated the utility of $Log_A\mathbf{G}$ in default reasoning, reasoning with information reported through a chain of sources, and even reasoning with paradoxical propositions. A careful examination of how $Log_A\mathbf{G}$ relates to other graded logics and non-monotonic formalisms is called for. On a first pass, we believe that $Log_A\mathbf{G}$ subsumes possibilistic logic [2], circumscription [9], and default theories with at most one justification per default (which includes normal defaults) [7]. We are currently working on the implementation of a proof theory for $Log_A\mathbf{G}$ based on a reason maintenance system (cf. [22]). The primary objective we have in mind is prioritized belief revision based on graded propositions.

References

1. Dubois, D., Godo, L., Prade, H.: Weighted logics for artificial intelligence—an introductory discussion. International Journal of Approximate Reasoning **55**(9) (2014) 1819–1829

2. Dubois, D., Prade, H.: Possibilistic logic: A retrospective and prospective view. Fuzzy Sets and Systems **144** (2004) 3–23
3. Halpern, J.: Reasoning About Uncertainty. MIT Press, Cambridge, MA (2005)
4. Hájek, P., Cintula, P.: On theories and models in fuzzy predicate logic. The Journal of Symbolic Logic **71**(3) (2006) 863–880
5. Smith, N.: Vagueness and Degrees of Truth. Oxford university Press, Oxford (2008)
6. Alchourron, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. The Journal of Symbolic Logic **50**(2) (1985) 510–530
7. Reiter, R.: A logic for default reasoning. Artificial Intelligence **13** (1980) 81–132
8. Kripke, S.: Outline of a theory of truth. Journal of Philosophy **72**(19) (1975) 690–716
9. McCarthy, J.: Circumscription— A form of non-monotonic reasoning. Artificial Intelligence **13** (1980) 27–39
10. Ismail, H.O.: $Log_A\mathbf{B}$: A first-order, non-paradoxical, algebraic logic of belief. Logic Journal of the IGPL **20**(5) (2012) 774–795
11. Ismail, H.O.: Stability in a commonsense ontology of states. In: Proceedings of the Eleventh International Symposium on Logical Formalization of Commonsense Reasoning (COMMONSENSE 2013), Agya Napa, Cyprus (2013)
12. Church, A.: On carnap's analysis of statements of assertion and belief. Analysis **10**(5) (1950) 97–99
13. Bealer, G.: Theories of properties, relations, and propositions. The Journal of Philosophy **76**(11) (1979) 634–648
14. Parsons, T.: On denoting propositions and facts. Philosophical Perspectives **7** (1993) 441–460
15. Shapiro, S.C.: Belief spaces as sets of propositions. Journal of Experimental and Theoretical Artificial Intelligence **5** (1993) 225–235
16. Demolombe, R.: Graded trust. In: Proceedings of the Workshop on Trust in Agent Societies (TRUST'09), Budapest, Hungary (2009)
17. Milošić, M., Obnjanović, Z.: A first-order conditional probability logic. Logic Journal of the IGPL **20**(1) (2012) 235–253
18. Burris, S., Sankappanavar, H.P.: A Course in Universal Algebra. Springer-Verlag (1982)
19. Hansson, S.O.: Kernel contraction. Journal of Symbolic Logic **59**(3) (1994) 845–859
20. Gupta, A., Belnap, N.: The Revision Theory of Truth. MIT Press, Cambridge, MA (1993)
21. Hájek, P., Paris, J., Shephedson, J.: The liar paradox and fuzzy logic. The Journal of Symbolic Logic **65**(1) (2000) 339–346
22. Ismail, H.O.: A reason maintenance perspective on relevant Ramsey conditionals. Logic Journal of the IGPL **18**(4) (2010) 508–529

A Proofs

A.1 Proof of Theorem 1

To prove Theorem 1, we need the following result.

Lemma 1. *If $\mathfrak{T}$ is a telescoping structure then, for every $n \in \mathbb{N}$, if $\mathfrak{F}^i(\mathfrak{T})$ is defined and $\mathfrak{F}^i(\mathfrak{T}) \cap Range(\mathfrak{g}) = \tau^i(\mathfrak{T}) \cap Range(\mathfrak{g})$ for every $i \leq n$, then $\mathfrak{F}^n(\mathfrak{T}) = F(\mathcal{Q})$, for some $\mathcal{Q} \subseteq E(\mathcal{T})$ and $\mathfrak{F}^n(\mathfrak{T}) \cap Range(\mathfrak{g}) \subseteq E^n(\mathcal{T})$.*

Proof. We prove the lemma by induction on n. For $n = 0$, $\mathfrak{F}^0(\mathcal{T}) = F(\mathcal{T})$, where $\mathcal{T} \subseteq E(\mathcal{T})$. Further, $\mathfrak{F}^0(\mathcal{T}) \cap Range(\mathfrak{g}) = \tau^0(\mathfrak{T}) \cap Range((g)) = \mathcal{T} \cap Range(\mathfrak{g}) \subseteq E^0(\mathcal{T})$. Now assume that the statement holds for some $k \in \mathbb{N}$. By the induction hypothesis, $\mathfrak{F}^k(\mathfrak{T}) \cap Range(\mathfrak{g}) \subseteq E^k(\mathcal{T})$. Therefore, all propositions graded in $E^1(\mathfrak{F}^k(\mathfrak{T}))$ are in $E^{k+1}(\mathcal{T})$. Moreover, $E^1(\mathfrak{F}^k(\mathfrak{T})) \cap Range(\mathfrak{g}) \subseteq E^{k+1}(\mathcal{T})$. By Observation 2, $\tau_{\mathfrak{T}}^{k+1}(\mathcal{T}) = \tau_{\mathfrak{T}}(\mathfrak{F}^k(\mathfrak{T})) = \varsigma(\kappa(E^1(\mathfrak{F}^k(\mathfrak{T}))), \mathfrak{T}), \mathcal{T}) = F(\mathcal{T}) \cup G$, where G is a set of propositions graded in $\kappa(E^1(\mathfrak{F}^k(\mathfrak{T})))$. By Definition 12, $\kappa(E^1(\mathfrak{F}^k(\mathfrak{T}))) \subseteq E^1(\mathfrak{F}^k(\mathfrak{T}))$. Hence, $G \subseteq E^{k+1}(\mathcal{T})$. Thus, $\mathfrak{F}^{k+1}(\mathfrak{T}) = F([F(\mathcal{T}) \cup G]) = F(\mathcal{T} \cup G)$, where $\mathcal{T} \cup G \subseteq E(\mathcal{T})$. Moreover, $\mathfrak{F}^{k+1}(\mathfrak{T}) \cap Range(\mathfrak{g}) = \tau^{k+1}(\mathfrak{T}) \cap Range(\mathfrak{g}) \subseteq E^{k+1}(\mathcal{T})$. □

We now proceed to proving the theorem. By Definition 9 and Clause 4 of Proposition 1, $E(\mathcal{T})$ has finitely-many grading propositions. In fact, since $\mathcal{T}$ is finite, then $E(\mathcal{T})$ is finite. Let $b = |E(\mathcal{T})|$. Now, taking $n = 2^b + 1$, suppose that, for every $i \leq 2^b + 1$, $\mathfrak{F}^i(\mathfrak{T})$ is defined and $\mathfrak{F}^i(\mathfrak{T}) \cap Range(\mathfrak{g}) = \tau^i(\mathfrak{T}) \cap Range(\mathfrak{g})$. By Lemma 1, $\mathfrak{F}^i(\mathfrak{T}) = F(\mathcal{Q}_i)$, for some $\mathcal{Q}_i \subseteq E(\mathcal{T})$. Since there are only 2^b subsets of $E(\mathcal{T})$, then $\mathfrak{F}^n(\mathfrak{T}) = \mathfrak{F}^j(\mathfrak{T})$, for some $j \leq 2^b$. Further, by Proposition 2, for every $j \in \mathbb{N}$, there is some $k \leq n$ such that $\mathfrak{F}^{n+j}(\mathfrak{T}) = \mathfrak{F}^k(\mathfrak{T})$. □

A.2 Proof of Corollary 1

We prove the case of $k = 1$ and the result follows by the same argument for all $k \in \mathbb{N}$.

$\mathfrak{F}^{n+1}(\mathfrak{T})$
$= F(\varsigma(\kappa(E^1(\mathfrak{F}^n(\mathfrak{T})), \mathfrak{T})), \mathcal{T})$ (Definitions 14 and 15)
$= F(\varsigma(\kappa(E^1(F(E(\mathcal{T}))))))$
$= F(\varsigma(\kappa(F(E(\mathcal{T})))))$ (Definition of E and the assumption of Theorem 1)
$= F(\varsigma(F(E(\mathcal{T}))))$ (Since $F(E(\mathcal{T}))$ is proper given that $\mathfrak{F}^{n+1}(\mathfrak{T})$ is defined)
$= F(F(E(\mathcal{T})))$ (Since every $p \in F(E(\mathcal{T}))$ is supported given that $F(E(\mathcal{T})) = \mathfrak{F}^n(\mathfrak{T})$ and Definitions 14 and 15)
$= F(E(\mathcal{T}))$

□

A.3 Proof of Theorem 2

For $n = 0$, the statement is trivial, since $\mathfrak{F}^0(\mathfrak{T}) = F(\mathcal{T})$. Otherwise, the statement follows directly from Observation 1 since, by Definition 15, $\mathfrak{F}^{k+1}(\mathfrak{T}) = F(K)$, for some $K \subseteq \kappa(E^1(\mathfrak{F}^k(\mathfrak{T})), \mathfrak{T})$. □

On the Functional Completeness of Argumentation Semantics*

Massimiliano Giacomin[1], Thomas Linsbichler[2], and Stefan Woltran[2]

[1] University of Brescia, Italy
[2] TU Wien, Austria

Abstract. Abstract argumentation frameworks (AFs) are one of the central formalisms in AI; equipped with a wide range of semantics, they have proven useful in several application domains. We contribute to the systematic analysis of semantics for AFs by connecting two recent lines of research – the work on input/output frameworks and the study of the expressiveness of semantics. We do so by considering the following question: given a function describing an input/output behaviour by mapping extensions (resp. labellings) to sets of extensions (resp. labellings), is there an AF with designated input and output arguments realizing this function under a given semantics? For the major semantics we give exact characterizations of the functions which are realizable in this manner.

1 Introduction

Dung's argumentation frameworks (AFs) have been extensively investigated, mainly because they represent an abstract model unifying a large variety of specific formalisms ranging from nonmonotonic reasoning to logic programming and game theory [9]. After the development and analysis of different semantics [13, 6, 2], recent attention has been drawn to their expressive power, i.e. determining which sets of extensions [10] and labellings [11] can be enforced in a single AF under a given semantics. Such results have recently been facilitated in order to express AGM-based revision in the context of abstract argumentation [8].

In [1], it has been shown that an AF can be viewed as a set of partial interacting sub-frameworks each characterized by an input/output behavior, i.e. a semantics-dependent function which maps each labelling of the "input" arguments (the external arguments affecting the sub-framework) into the set of labellings prescribed for the "output" arguments (the arguments of the sub-framework affecting the external ones). It turns out that under the major semantics, i.e. complete, grounded, stable and (under some mild conditions) preferred semantics, sub-frameworks with the same input/output behavior can be safely exchanged, i.e. replacing a sub-framework with an equivalent one does not affect the justification status of the invariant arguments: semantics of this kind are called *transparent* [1].

* This research has been supported by the Austrian Science Fund (FWF) through projects I1102 and P25521.

Then, somewhat resembling functional completeness of a specific set of logic gates, a natural question concerns the expressive power of transparent semantics in the context of an interacting sub-framework: given a so-called *I/O specification*, i.e. a function describing an input/output behaviour by mapping extensions (resp. labellings) to sets of extensions (resp. labellings), is there an AF with designated input and output arguments realizing this function under a given semantics? In this paper, we answer this question as follows:

– For the stable, preferred, semi-stable, stage, complete, ideal, and grounded semantics we exactly characterize all realizable two-valued I/O specifications.
– For the preferred and grounded labellings we exactly characterize all realizable three-valued I/O specifications.

Answering this question is essential in many aspects. First, it adds to the analysis and *comparison of semantics* (see e.g. [5, 3]), by providing an absolute characterization of their functional expressiveness, which holds independently of how the abstract argumentation framework is instantiated. Second, it lays foundations towards a theory of *dynamic and modular argumentation*. More specifically, a functional characterization provides a common ground for different representations of the same sub-framework, e.g. to devise a summarized version of a sub-framework, or to give an argumentation-based view of the same framework at different levels of abstraction, as in metalevel argumentation [12]. One may also translate a different formalism to an AF or vice versa, or provide an argument-free representation of a given AF for human/computer interaction issues: in all of these cases, it is important to know whether an input/output behavior is realizable under a given argumentation semantics. Finally, our results are important in the dynamic setting of *strategic argumentation*, where a player may exploit the fact that for some set of arguments certain labellings are achievable (or non achievable) independently of the labelling of other arguments, or more generally she/he may exploit knowledge on the set of realizable dependencies. For example, an agent may desire to achieve some goal, i.e., ensure that a certain argument is justified. Considering arguments brought up by other agents as input arguments, our results enable to verify whether the goal is achievable and provide one particular way for the agent to bring up further arguments in order to succeed.

The paper is organized as follows. After providing the necessary background in Section 2, Section 3 introduces the notion of *I/O*-gadget to represent a sub-framework, and tackles the above problem with extension-based two-valued specifications. Labelling-based three-valued specifications are investigated in Section 4 and Section 5 concludes the paper.

2 Background

We assume a countably infinite domain of arguments $\mathfrak{A}$. An *argumentation framework* (AF) is a pair $F = (A, R)$ where $A \subseteq \mathfrak{A}$ and $R \subseteq A \times A$. We assume that A is non-empty and finite. For an AF $F = (A, R)$ and a set of arguments

$S \subseteq A$, we define $S_F^+ = \{a \in A \mid \exists s \in S : (s,a) \in R\}$, $S_F^\oplus = S \cup S_F^+$, and $S_F^- = \{a \in A \mid \exists s \in S : (a,s) \in R\}$.

Given $F = (A,R)$, a set $S \subseteq A$ is *conflict-free* (in F), if there are no arguments $a, b \in S : (a,b) \in R$. An argument $a \in A$ is *defended* (in F) by a set $S \subseteq A$ if $\forall b \in A : (b,a) \in R \Rightarrow b \in S_F^+$. A set $S \subseteq A$ is *admissible* (in F) if it is conflict-free and defends all of its elements. We denote the set of conflict-free and admissible sets in F as $cf(F)$ and $ad(F)$, respectively.

An extension-based semantics σ associates to any $F = (A,R)$ the (possibly empty) set $\sigma(F)$ including subsets of A called σ-extensions. In this paper we focus on complete, grounded, preferred, ideal, stable, stage and semi-stable semantics, with extensions defined as follows:

- $S \in co(F)$ iff $S \in ad(F)$ and $a \in S$ for all $a \in A$ defended by S;
- $S \in gr(F)$ iff S is the least element in $co(F)$;
- $S \in pr(F)$ iff $S \in ad(F)$ and $\nexists T \in ad(F)$ s.t. $T \supset S$;
- $S \in id(F)$ iff $S \in ad(F)$, $S \subseteq \bigcap pr(F)$ and $\nexists T \in ad(F)$ s.t. $T \subseteq \bigcap pr(F)$ and $T \supset S$;
- $S \in st(F)$ iff $S \in cf(F)$ and $S_F^\oplus = A$;
- $S \in sg(F)$ iff $S \in cf(F)$ and $\nexists T \in cf(F)$ s.t. $T_F^\oplus \supset S_F^\oplus$;
- $S \in se(F)$ iff $S \in ad(F)$ and $\nexists T \in ad(F)$ s.t. $T_F^\oplus \supset S_F^\oplus$.

Given $F = (A,R)$ and a set $O \subseteq A$, the restriction of $\sigma(F)$ to O, denoted as $\sigma(F)|_O$, is the set $\{E \cap O \mid E \in \sigma(F)\}$.

Given a set of arguments A, a labelling L is a function assigning each argument $a \in A$ exactly one label among `t`, `f` and `u`, i.e. $L : A \mapsto \{\mathtt{t}, \mathtt{f}, \mathtt{u}\}$. If the arguments $A = \{a_1, \ldots, a_n\}$ are ordered, then we denote a labelling of A as a sequence of labels, e.g. the labelling `tuf` of arguments $\{a_1, a_2, a_3\}$ maps a_1 to `t`, a_2 to `u`, and a_3 to `f`. We denote the set of all possible labellings of A as $\mathcal{L}(A)$. Likewise, given an AF F, we denote the set of all possible labellings of (the arguments of) F as $\mathcal{L}(F)$. Given a labelling L and an argument a, $L(a)$ denotes the labelling of a wrt. L; finally $in(L)$, $out(L)$, and $undec(L)$ denotes the arguments labeled to `t`, `f`, and `u` by L, respectively.

Definition 1. *Given a set of arguments A and labellings L_1, L_2 thereof, $L_1 \sqsubseteq L_2$ iff $in(L_1) \subseteq in(L_2)$ and $out(L_1) \subseteq out(L_2)$. Moreover we call L_1 and L_2*

- comparable *if $L_1 \sqsubseteq L_2$ or $L_2 \sqsubseteq L_1$*
- compatible *if $in(L_1) \cap out(L_2) = out(L_1) \cap in(L_2) = \emptyset$*

Note that if L_1 and L_2 are comparable then they are also compatible.

An argumentation semantics can be defined in terms of labellings rather than of extensions, i.e. the labelling-based version of a semantics σ associates to F a set $\mathcal{L}_\sigma(F) \subseteq \mathcal{L}(F)$, where any labelling $L \in \mathcal{L}_\sigma(F)$ corresponds to an extension $S \in \sigma(F)$ as follows: an argument $a \in A$ is labeled to `t` iff $a \in S$, is labeled to `f` iff $a \in S_F^+$, is labeled to `u` if neither of the above conditions holds. Given F and a set $O \subseteq A$, the restriction of $\mathcal{L}_\sigma(F)$ to O, denoted as $\mathcal{L}_\sigma(F)|_O$, is the set $\{L \cap (O \times \{\mathtt{t}, \mathtt{f}, \mathtt{u}\}) \mid L \in \mathcal{L}_\sigma(F)\}$. The following well-known result can be deduced e.g. from the semantics account given in [7].

Proposition 1. *For an AF F, $\mathcal{L}_{pr}(F) = \max_\sqsubseteq(\mathcal{L}_{co}(F))$.*

3 Extension-based I/O-gadgets

An I/O-gadget represents a (partial) AF where two sets of arguments are identified as input and output arguments, respectively, with the restriction that input arguments do not have any ingoing attacks.

Definition 2. *Given a set of input arguments $I \subseteq \mathfrak{A}$ and a set of output arguments $O \subseteq \mathfrak{A}$ with $I \cap O = \emptyset$, an I/O-*gadget *is an* AF $F = (A, R)$ *such that* $I, O \subseteq A$ *and* $I_F^- = \emptyset$.

The injection of a set $J \subseteq I$ to F simulates the input J in the way that all arguments in J are accepted (none of them has ingoing attacks since F is an I/O-gadget) and all arguments in $(I \setminus J)$ are rejected (each of them is attacked by the newly introduced argument z, which has no ingoing attacks).

Definition 3. *Given an I/O-gadget $F = (A, R)$ and a set of arguments $J \subseteq I$, the injection of J to F is the* AF $\triangleright(F, J) = (A \cup \{z\}, R \cup \{(z, i) \mid i \in (I \setminus J)\})$, *where z is a newly introduced argument.*

An I/O-specification describes a desired input/output behaviour by assigning to each set of input arguments a set of sets of output arguments.

Definition 4. *A* two-valued[3] I/O-specification *consists of two sets $I, O \subseteq \mathfrak{A}$ and a total function $\mathfrak{f} : 2^I \mapsto 2^{2^O}$.*

In order for an I/O-gadget F to satisfy $\mathfrak{f}$, the injection of each $J \subseteq I$ to F must have $\mathfrak{f}(J)$ as its σ-extensions restricted to the output arguments. So, informally, with input J applied the set of outputs under σ should be exactly $\mathfrak{f}(J)$.

Definition 5. *Given $I, O \subseteq \mathfrak{A}$, a semantics σ and an I/O-specification $\mathfrak{f}$, the I/O-gadget F satisfies $\mathfrak{f}$ under σ iff $\forall J \subseteq I : \sigma(\triangleright(F, J))|_O = \mathfrak{f}(J)$.*

The question we want to address is which conditions $\mathfrak{f}$ must fulfill to be satisfiable by some I/O-gadget and how an I/O-gadget doing so can be constructed.

Definition 6. *Given an I/O-specification $\mathfrak{f}$, let $Y = \{y_i \mid i \in I\}$ and $X = \{x_J^S \mid J \subseteq I, S \in \mathfrak{f}(J)\}$. The* canonical I/O-gadget *is defined as*

$$\begin{aligned}
\mathfrak{C}(\mathfrak{f}) = (&I \cup O \cup Y \cup X \cup \{w\},\\
&\{(i, y_i) \mid i \in I\} \cup\\
&\{(y_i, x_J^S) \mid x_J^S \in X, i \in J\} \cup\\
&\{(i, x_J^S) \mid x_J^S \in X, i \in (I \setminus J)\} \cup\\
&\{(x_1, x_2) \mid x_1, x_2 \in X, x_1 \neq x_2\} \cup\\
&\{(x_i, w) \mid x_i \in X\} \cup \{(w, w)\} \cup\\
&\{(x_J^S, o) \mid J \subseteq I, S \in \mathfrak{f}(J), o \in (O \setminus S)\}).
\end{aligned}$$

[3] In the following we omit this specification.

Intuitively, x_J^S shall enforce output S for input J. Moreover, w ensures that any stable extension of (an injection to) $\mathfrak{C}(\mathfrak{f})$ must contain an argument in X.

The following theorem shows that any I/O-specification is satisfiable under stable semantics.

Theorem 1. *Every I/O-specification $\mathfrak{f}$ is satisfied by $\mathfrak{C}(\mathfrak{f})$ under st.*

Proof. Let $I, O \subseteq \mathfrak{A}$ and $\mathfrak{f}$ be an arbitrary I/O-specification. We have to show that $st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))|_O = \mathfrak{f}(J)$ holds for any $J \subseteq I$. Consider such a $J \subseteq I$.

First let $S \in \mathfrak{f}(J)$. We show that $E = \{z\} \cup J \cup \{y_i \mid i \in (I \setminus J)\} \cup \{x_J^S\} \cup S \in st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))$, thus $S \in st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))|_O$. E is conflict-free in $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$ since z only attacks the arguments $(I \setminus J)$, an y_i with $i \in (I \setminus J)$ is only attacked by $i \notin E$, x_J^S is only attacked by other $x \in X$, $i \in (I \setminus J)$ and y_j with $j \in J$, and arguments in S are only attacked by arguments from X but not from x_J^S. E is stable in $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$ since x_J^S attacks w, all other $x \in X$ and all $o \in (O \setminus S)$; z attacks all $i \in (I \setminus J)$; each y_j with $j \in J$ is attacked by j.

It remains to show that there is no $S' \in st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))|_O$ with $S' \notin \mathfrak{f}(J)$. Towards a contradiction assume there is some $S' \in st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))|_O$ with $S' \notin \mathfrak{f}(J)$. Hence there must be some $E' \in st(\triangleright(\mathfrak{C}(\mathfrak{f}), J))$ with $S' \subset E'$. Since w attacks itself, $w \notin E'$, thus by construction of $\mathfrak{C}(\mathfrak{f})$ there must be some $x_{J'}^{S'} \in (X \cap E')$ attacking w, and $x_{J'}^{S'}$ must attack all $o \in (O \setminus S')$. Since $S' \notin \mathfrak{f}(J)$ by assumption, it must hold that $J' \neq J$. Now note that $z \in E'$ and $j \in E'$ for all $j \in J$, since they are not attacked by construction of $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$. Now if $J' \subset J$ then there is some $j \in (J \setminus J')$ attacking $x_{J'}^{S'}$, a contradiction to conflict-freeness of E'. On the other hand if $J' \not\subseteq J$ there is some $j' \in (J' \setminus J)$ which is attacked by z. Therefore also $y_{j'} \in E'$, which attacks $x_{J'}^{S'}$, again a contradiction. $\square$

As to stage, preferred and semi-stable semantics, any I/O-specification is satisfiable, provided that a (possibly empty) output is prescribed for any input.

Proposition 2. *Every I/O-specification $\mathfrak{f}$ such that $\forall J \subseteq I, \mathfrak{f}(J) \neq \emptyset$ is satisfied by $\mathfrak{C}(\mathfrak{f})$ under $\sigma \in \{sg, pr, se\}$.*

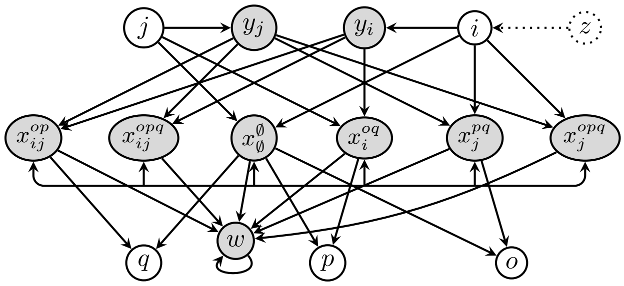
Proof (Sketch). Note that $\forall J \subseteq I$, stable, preferred, stage and semi-stable extensions coincide in $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$, thus the result follows from Theorem 1. $\square$

Theorem 2. *An I/O-specification $\mathfrak{f}$ is satisfiable under $\sigma \in \{sg, pr, se\}$ iff $\forall J \subseteq I, \mathfrak{f}(J) \neq \emptyset$.*

Proof. $\Leftarrow$: by Proposition 2.
$\Rightarrow$: Follows directly by the fact that in any AF, particularly in any injection of some extension to an I/O-gadget, a σ-extension always exists. $\square$

Example 1. Consider the I/O-specification $\mathfrak{f}$ with $I = \{i, j\}$ and $O = \{o, p, q\}$ defined as follows: $\mathfrak{f}(\emptyset) = \{\emptyset\}$; $\mathfrak{f}(\{i\}) = \{\{o, q\}\}$; $\mathfrak{f}(\{j\}) = \{\{o, p, q\}, \{p, q\}\}$; and $\mathfrak{f}(\{i, j\}) = \{\{o, p, q\}, \{o, p\}\}$. The canonical I/O-gadget $\mathfrak{C}(\mathfrak{f})$ is depicted below[4] (without the dotted part):

[4] Argument names such as $x_{\{a,b\}}^{\{c,d\}}$ are abbreviated by x_{ab}^{cd}.



Let σ be a semantics in $\{st, sg, se, pr\}$. One can verify that for every possible input $J \subseteq I$, the injection of J to $\mathfrak{C}(\mathfrak{f})$, has exactly $\mathfrak{f}(J)$ as σ-extensions restricted to O. As an example let $J = \{j\}$. $\triangleright(\mathfrak{C}(\mathfrak{f}), \{j\})$ adds the argument z attacking i to $\mathfrak{C}(\mathfrak{f})$. Now $\sigma(\triangleright(\mathfrak{C}(\mathfrak{f}), \{j\})) = \{\{z, j, y_i, x_{\{j\}}^{\{p,q\}}, p, q\}, \{z, j, y_i, x_{\{j\}}^{\{o,p,q\}}, o, p, q\}\}$, hence $\sigma(\triangleright(\mathfrak{C}(\mathfrak{f}), \{j\}))|_O = \{\{p, q\}, \{o, p, q\}\} = \mathfrak{f}(\{j\})$.

Also for complete, grounded and ideal semantics we are able to identify a necessary and sufficient condition for satisfiability. While we show sufficiency of these conditions in more detail, their necessity is by the well-known facts that the intersection of all complete extensions is always a complete extension too and ideal and grounded semantics always yield exactly one extension.

Definition 7. *An I/O-specification $\mathfrak{f}$ is* closed *iff for each $J \subseteq I$ it holds that $\mathfrak{f}(J) \neq \emptyset$ and $\bigcap \mathfrak{f}(J) \in \mathfrak{f}(J)$.*

Proposition 3. *Every closed I/O-specification $\mathfrak{f}$ is satisfied by $\mathfrak{C}(\mathfrak{f})$ under co.*

Proof. Let $J \subseteq I$ and $\mathbb{S} = \mathfrak{f}(J)$. By construction of $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$, $E^* = \{z\} \cup J \cup \{y_i \mid i \in (I \setminus J)\}$ is contained in all complete extensions, while the elements of $(I \setminus J) \cup \{y_i \mid i \in J\}$ are attacked by E^* and thus by all complete extensions. All $x_{J'}^{S'}$ with $J' \neq J$ are attacked by J or some y_i with $i \in (I \setminus J)$, thus they are attacked by E^*, while all x_J^S with $S \in \mathbb{S}$ attack each other and the other attacks they receive come from elements attacked by E^* in turn. Two cases can then be distinguished. If $|\mathbb{S}| = 1$ then by construction of $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$ there is just one x_J^S defended by E^*, thus the only complete extension is $E^* \cup \{x_J^S\} \cup S$. If, on the other hand, $|\mathbb{S}| > 1$, any x_J^S with $S \in \mathbb{S}$ can be included, giving rise to the complete extension $E^* \cup \{x_J^S\} \cup S$, or none of x_J^S can be included, giving rise to the complete extension $E^* \cup \bigcap \mathbb{S}$ since an x_J^S attacks all $o \in (O \setminus S)$. Taking into account that $\bigcap \mathbb{S} \in \mathbb{S}$, in both cases we have that $co(\triangleright(F, J))|_O = \mathfrak{f}(J)$. □

Theorem 3. *An I/O-specification $\mathfrak{f}$ is satisfiable under co iff $\mathfrak{f}$ is closed.*

Proposition 4. *Every I/O-specification $\mathfrak{f}$ with $|\mathfrak{f}(J)| = 1$ for each $J \subseteq I$ is satisfied by $\mathfrak{C}(\mathfrak{f})$ under gr and id.*

Proof (Sketch). This follows the same idea as the proof of Proposition 3. Since $|\mathfrak{f}(J)| = 1$, $\triangleright(\mathfrak{C}(\mathfrak{f}), J)$ has only one complete extension for each $J \subseteq I$, which is also grounded and ideal. □

Theorem 4. *An I/O-specification $\mathfrak{f}$ is satisfiable under gr and id iff $|\mathfrak{f}(J)| = 1$ for each $J \subseteq I$.*

4 Labelling-based I/O-gadgets

In the previous section we have dealt with I/O-specifications mapping extensions to sets of extensions. In general, there are two reasons why an argument does not belong to an extension, i.e. either because it is attacked by the extension or because it is undecided due to insufficient justification. This distinction impacts on the justification status of arguments, since attacks from undecided arguments can prevent attacked arguments to belong to an extension, while attacks from out arguments are ineffective. In order to take into account this distinction, we first provide a labelling-based counterpart of the notions introduced in Section 3.

Definition 8. *A* 3-valued I/O-specification *consists of two sets $I, O \subseteq \mathfrak{A}$ and a total function $\mathfrak{f} : \mathcal{L}(I) \mapsto 2^{\mathcal{L}(O)}$.*

Definition 9. *Given an I/O-gadget $F = (A, R)$ and a labelling L of I, the labelling-injection of L to F is the* AF $\blacktriangleright(F, L) = (A \cup \{z\}, R \cup \{(z, a) \mid L(a) = f\} \cup \{(b, b) \mid L(b) = \mathtt{u}\})$*, where z is a newly introduced argument.*

Definition 10. *Given $I, O \subseteq \mathfrak{A}$, a semantics σ and a 3-valued I/O-specification $\mathfrak{f}$, the I/O-gadget F satisfies $\mathfrak{f}$ under σ iff $\forall L \in \mathcal{L}(I) : \mathcal{L}_\sigma(\blacktriangleright(F, L))|_O = \mathfrak{f}(L)$.*

By definition of stable labellings it is clear that in order to be satisfied under *st*, a 3-valued I/O-specification must only be non-empty for labellings with each argument labeled to `t` or `f`.

Theorem 5. *A 3-valued I/O-specification $\mathfrak{f}$ is satisfiable under st iff for each $L \in \mathcal{L}(I)$ it holds that*

- *if $\exists i \in I : L(i) = \mathtt{u}$ then $\mathfrak{f}(L) = \emptyset$, and*
- *otherwise $K(o) \neq \mathtt{u}$ for all $K \in \mathfrak{f}(L)$ and $o \in O$.*

Proof. $\Rightarrow$: If $\exists i \in I : L(i) = \mathtt{u}$ then for any I/O-gadget F, $\blacktriangleright(F, L)$ contains a self-attacking argument otherwise unattacked, hence $\mathcal{L}_{st}(\blacktriangleright(F, L)) = \emptyset$. In the other case, by definition of stable extension it is clear that each $o \in O$ must be labelled either `t` or `f` by any stable labelling.
$\Leftarrow$: Follows from Theorem 1. If $\exists i \in I : L(i) = \mathtt{u}$ then $st(\blacktriangleright(\mathfrak{C}(\mathfrak{f}), L)) = \emptyset$, otherwise the labelling-injection coincides with the normal injection. ◻

In order to characterize those 3-valued I/O-specifications which are satisfiable under the other semantics we need the concept of monotonicity.

Definition 11. *A 3-valued I/O-specification $\mathfrak{f}$ is* monotonic *iff for all L_1 and L_2 such that $L_1 \sqsubseteq L_2$ it holds that $\forall K_1 \in \mathfrak{f}(L_1) \exists K_2 \in \mathfrak{f}(L_2) : K_1 \sqsubseteq K_2$.*

The intuitive meaning of monotonicity is the following: if K_1 is an output for input L_1, then for every input which is more informative than L_1 there must be an output more informative than K_1. First a rather obvious observation:

Proposition 5. *For every 3-valued I/O-specification $\mathfrak{f}$ which is satisfiable under gr, $|\mathfrak{f}(L)| = 1$ for all $L \in \mathcal{L}(I)$.*

The following was shown in Proposition 7 of [1] for complete semantics.

Proposition 6. *Every* 3*-valued* I/O*-specification which is satisfiable under* $\{gr, pr\}$ *is monotonic.*

Proof (Sketch). Let $\mathfrak{f}$ be a 3-valued I/O-specification and suppose it is satisfied by the I/O-gadget F under gr. Moreover let $L_1 \sqsubseteq L_2$ be labellings of I. Proposition 7 of [1] says that $\forall K_2 \in \mathcal{L}_{co}(\blacktriangleright(F, L_2))|_O \, \exists K_1 \in \mathcal{L}_{co}(\blacktriangleright(F, L_1))|_O : K_1 \sqsubseteq K_2$ and $\forall K_1 \in \mathcal{L}_{co}(\blacktriangleright(F, L_1))|_O \, \exists K_2 \in \mathcal{L}_{co}(\blacktriangleright(F, L_2))|_O : K_1 \sqsubseteq K_2$. By the facts that $|\mathfrak{f}(L_1)|$=$|\mathfrak{f}(L_2)|$=1 (cf. Proposition 5) and $\mathcal{L}_{pr}(F) = \max_{\sqsubseteq}(\mathcal{L}_{co}(F))$ for each AF F (cf. Proposition 1), the result follows for gr and pr, respectively. $\square$

In Propositions 5 and 6 we have given necessary conditions for 3-valued I/O-specifications to be satisfiable under gr and pr. In the following we show that these conditions are also sufficient in the sense that we can find a satisfying I/O-gadget. The constructions of these I/O-gadgets will depend on the given 3-valued I/O-specification and on the semantics, but they will share the same input and output part. The semantics-specific part, denoted by $X^\sigma_{\mathfrak{f}}$ and $R^\sigma_{\mathfrak{f}}$ in the following definition, will be given later.

Definition 12. *Given a* 3*-valued* I/O*-specification* $\mathfrak{f}$ *we define* $I' = \{i' \mid i \in I\}$, $O' = \{o' \mid o \in O\}$, $R_I = \{(i, i') \mid i \in I\}$ *and* $R_O = \{(o', o'), (o', o) \mid o \in O\}$*. The* 3-valued canonical I/O-gadget *for semantics* σ *is defined as*

$$\mathfrak{D}^\sigma_{\mathfrak{f}} = (I \cup I' \cup X^\sigma_{\mathfrak{f}} \cup O' \cup O, R_I \cup R^\sigma_{\mathfrak{f}} \cup R_O).$$

with $R^\sigma_{\mathfrak{f}} \subseteq ((I \cup I') \times X^\sigma_{\mathfrak{f}}) \cup (X^\sigma_{\mathfrak{f}} \times X^\sigma_{\mathfrak{f}}) \cup (X^\sigma_{\mathfrak{f}} \times (O' \cup O))$.

Now we turn to the semantics-specific constructions. For grounded semantics we need the concept of *determining* input labellings. An input labelling L is determining for output argument o if L is a minimal (w.r.t. $\sqsubseteq$) input labelling where o gets a concrete value ($\mathtt{t}$ or $\mathtt{f}$) according to $\mathfrak{f}$.

With abuse of notation, in the following we may identify a set including a single labelling with the labelling itself.

Definition 13. *Given a* 3*-valued* I/O*-specification* $\mathfrak{f}$ *with* $|\mathfrak{f}(L)| = 1$ *for all* $L \in \mathcal{L}(I)$ *and an argument* $o \in O$*, a labelling* L *of* I *is determining for* o*, if* $\mathfrak{f}(L)(o) \neq \mathtt{u}$ *and* $\forall L' \sqsubseteq L : \mathfrak{f}(L')(o) = \mathtt{u}$*. We denote the set of labellings which are determining for* o *as* $\mathfrak{d}_{\mathfrak{f}}(o)$*.*

Example 2. Let $\mathfrak{f}$ be the following 3-valued I/O-specification with $I = \{i_1, i_2\}$ and $O = \{o_1, o_2\}$: $\mathfrak{f}(\mathtt{uu}) = \{\mathtt{uu}\}$; $\mathfrak{f}(\mathtt{tu}) = \{\mathtt{tu}\}$; $\mathfrak{f}(\mathtt{ut}) = \{\mathtt{ut}\}$; $\mathfrak{f}(\mathtt{uf}) = \{\mathtt{uf}\}$; $\mathfrak{f}(\mathtt{fu}) = \{\mathtt{uu}\}$; $\mathfrak{f}(\mathtt{tt}) = \{\mathtt{tt}\}$; $\mathfrak{f}(\mathtt{tf}) = \{\mathtt{tf}\}$; $\mathfrak{f}(\mathtt{ft}) = \{\mathtt{ut}\}$; and $\mathfrak{f}(\mathtt{ff}) = \{\mathtt{tf}\}$ We have the following sets of determining labellings: $\mathfrak{d}_{\mathfrak{f}}(o_1) = \{\mathtt{tu}, \mathtt{ff}\}$ and $\mathfrak{d}_{\mathfrak{f}}(o_2) = \{\mathtt{ut}, \mathtt{uf}\}$. Consider, for instance, the input labelling $\mathtt{ff}$. We have $\mathfrak{f}(\mathtt{ff}) = \mathtt{tf}$. In order to check if $\mathtt{ff}$ is determining for o_1 we have to look at all input labellings being less committed than $\mathtt{ff}$. Now we observe $\mathfrak{f}(\mathtt{uf}) = \mathtt{uf}$, $\mathfrak{f}(\mathtt{fu}) = \mathfrak{f}(\mathtt{uu}) = \mathtt{uu}$. In all of these desired output labellings o_1 has value $\mathtt{u}$, so $\mathtt{ff}$ is determining for o_1. On the other hand $\mathtt{ff}$ is not determining for o_2, since $\mathfrak{f}(\mathtt{uf})(o_2) = \mathtt{f}$.

Definition 14. *Given a 3-valued I/O-specification $\mathfrak{f}$ with $|\mathfrak{f}(L)=1|$ for all $L \in \mathcal{L}(I)$, the gr-specific part of $\mathfrak{D}_{\mathfrak{f}}^{gr}$ is given by*

$$X_{\mathfrak{f}}^{gr} = \{x_o^L \mid o \in O, L \in \mathfrak{d}_{\mathfrak{f}}(o)\}, \text{ and}$$
$$R_{\mathfrak{f}}^{gr} = \{(i, x_o^L) \mid x_o^L \in X_{\mathfrak{f}}^{gr}, L(i) = \mathtt{f}\} \cup \{(i', x_o^L) \mid x_o^L \in X_{\mathfrak{f}}^{gr}, L(i) = \mathtt{t}\} \cup$$
$$\{(x_o^L, o') \mid x_o^L \in X_{\mathfrak{f}}^{gr}, \mathfrak{f}(L)(o) = \mathtt{t}\} \cup \{(x_o^L, o) \mid x_o^L \in X_{\mathfrak{f}}^{gr}, \mathfrak{f}(L)(o) = \mathtt{f}\}.$$

For every $o \in O$ and each labelling L which is determining for o, there is the argument x_o^L. This argument can be labelled $\mathtt{t}$ if L is the labelling of I and intuitively enforces the labelling of o to be as given by $\mathfrak{f}(L)$.

The next results, requiring two preliminary lemmata, characterize satisfiability of grounded semantics.

Lemma 1. *Let $\mathfrak{f}$ be a 3-valued I/O-specification which is monotonic and s.t. $|\mathfrak{f}(L)| = 1$ for each $L \in \mathcal{L}(I)$. Let $o \in O$ and $L, L' \in \mathcal{L}(I)$ such that $\mathfrak{f}(L)(o) = \mathtt{t}$ and $\mathfrak{f}(L')(o) = \mathtt{f}$. Then L and L' are not compatible.*

Lemma 2. *Given a 3-valued I/O-specification $\mathfrak{f}$ which is monotonic and s.t. $|\mathfrak{f}(L)| = 1$ for each $L \in \mathcal{L}(I)$, let $o \in O$ and $L, L' \in \mathcal{L}(I)$ such that L is determining for o. Then $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L'))(x_o^L)$ is*

- *$\mathtt{t}$ iff $L \sqsubseteq L'$;*
- *$\mathtt{f}$ iff L and L' are not compatible; and*
- *$\mathtt{u}$ iff $L \not\sqsubseteq L'$ but L and L' are compatible.*

Proposition 7. *Every 3-valued I/O-specification $\mathfrak{f}$ which is monotonic and s.t. $|\mathfrak{f}(L)| = 1$ for each $L \in \mathcal{L}(I)$, is satisfied by $\mathfrak{D}_{\mathfrak{f}}^{gr}$ under gr.*

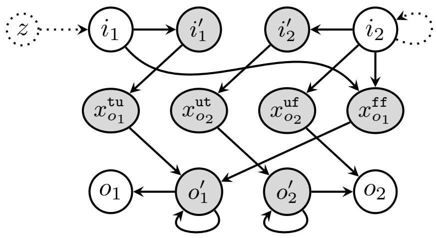
Proof. Consider some input labelling L. We have to show $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))|_O = \mathfrak{f}(L)$. To this end let $o \in O$.

Assume $\mathfrak{f}(L)(o) = \mathtt{u}$. Then, since $\mathfrak{f}$ is monotonic, $\mathfrak{f}(L')(o) = \mathtt{u}$ for all $L' \sqsubseteq L$. Therefore, there is no $L' \sqsubseteq L$ with $L' \in \mathfrak{d}_{\mathfrak{f}}(o)$. By Lemma 2 we get that for all $L'' \in \mathfrak{d}_{\mathfrak{f}}(o)$ it holds that $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(x_o^{L''}) \neq \mathtt{t}$. Since such $x_o^{L''}$ are the only potential attackers of o and o', $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(o) = \mathtt{u}$.

Next assume $\mathfrak{f}(L)(o) = \mathtt{t}$. Then there is some $L' \sqsubseteq L$ with $L' \in \mathfrak{d}_{\mathfrak{f}}(o)$ and $\mathfrak{f}(L')(o) = \mathtt{t}$. By Lemma 2 we get $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(x_o^{L'}) = \mathtt{t}$. Moreover, $x_o^{L'}$ attacks o', hence $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, K))(o') = \mathtt{f}$. Towards a contradiction assume there is some $x_o^{L''}$ attacking o with $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(x_o^{L''}) \in \{\mathtt{t}, \mathtt{u}\}$. Then, by Lemma 2, L'' and L are compatible. However, by construction of $\mathfrak{D}_{\mathfrak{f}}^{gr}$, $\mathfrak{f}(L'')(o) = \mathtt{f}$, $\mathfrak{f}(L)(o) = \mathtt{t}$ and, by Lemma 1 L'' and L are not compatible.

Finally assume $\mathfrak{f}(L)(o) = \mathtt{f}$. Then there is some $L' \sqsubseteq L$ with $L' \in \mathfrak{d}_{\mathfrak{f}}(o)$ and $\mathfrak{f}(L')(o) = \mathtt{f}$. By Lemma 2 we get $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(x_o^{L'}) = \mathtt{t}$. Moreover, $x_o^{L'}$ attacks o, hence $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L))(o) = \mathtt{f}$. □

Example 3. Again consider the 3-valued I/O-specification $\mathfrak{f}$ from Example 2. We have seen the determining labellings there. The I/O-gadget $\mathfrak{D}_{\mathfrak{f}}^{gr}$ is depicted below. Consider for example the labelling-injection of $\mathtt{fu}$ to $\mathfrak{D}_{\mathfrak{f}}^{gr}$, which is indicated by the dotted part of the figure. We get $\mathcal{L}_{gr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, \mathtt{fu}))|_O = \mathtt{uu}$, satisfying the I/O-specification. One can check that this holds for all possible labelling-injections, hence $\mathfrak{D}_{\mathfrak{f}}^{gr}$ satisfies $\mathfrak{f}$ under the grounded semantics.



Theorem 6. *A 3-valued I/O-specification $\mathfrak{f}$ is satisfiable under gr iff $\mathfrak{f}$ is monotonic and for each $L \in \mathcal{L}(I)$, $|\mathfrak{f}(L)| = 1$.*

Definition 15. *Given a 3-valued I/O-specification $\mathfrak{f}$, the pr-specific part of $\mathfrak{D}_{\mathfrak{f}}^{pr}$ is given by*

$$X_{\mathfrak{f}}^{pr} = \{x_K^L \mid L \in \mathcal{L}(I), K \in \mathfrak{f}(L)\}, \text{ and}$$
$$R_{\mathfrak{f}}^{pr} = \{(i, x_K^L) \mid x_K^L \in X_{\mathfrak{f}}^{pr}, L(i) = \mathit{f}\} \cup \{(i', x_K^L) \mid x_K^L \in X_{\mathfrak{f}}^{pr}, L(i) = \mathit{t}\} \cup$$
$$\{(x_K^L, o') \mid x_K^L \in X_{\mathfrak{f}}^{pr}, K(o) = \mathit{t}\} \cup \{(x_K^L, o) \mid x_K^L \in X_{\mathfrak{f}}^{pr}, K(o) = \mathit{f}\} \cup$$
$$\{(x_K^L, x_{K'}^{L'}) \mid \neg(L \sqsubset L' \wedge K \sqsubseteq K') \wedge \neg(L \sqsupset L' \wedge K \sqsupseteq K')\}.$$

Every input-output-combination is represented by an argument in $\mathfrak{D}_{\mathfrak{f}}^{pr}$. We first show a technical lemma, giving sufficient conditions on the labelling-status of the arguments in $X_{\mathfrak{f}}^{pr}$ to get the desired labelling of the output arguments.

Lemma 3. *Given a 3-valued I/O-specification $\mathfrak{f}$ and an input labelling $L \in \mathcal{L}(I)$. The following holds for each preferred labelling P of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$: If $P(x_K^L) = \mathit{t}$ and for all $x_{K'}^{L'} \in X_{\mathfrak{f}}^{pr}$, $K' \sqsupset K \Rightarrow P(x_{K'}^{L'}) \neq \mathit{t}$ and $(K' \not\sqsubseteq K \wedge K \not\sqsubseteq K') \Rightarrow P(x_{K'}^{L'}) = \mathit{f}$, then $P|_O = K$.*

We proceed by showing that $\mathfrak{D}^{pr}$ satisfies every monotonic function under the preferred semantics.

Proposition 8. *Every 3-valued I/O-specification $\mathfrak{f}$ which is monotonic is satisfied by $\mathfrak{D}_{\mathfrak{f}}^{pr}$ under pr.*

Proof. Consider an arbitrary input labelling $L \in \mathcal{L}(I)$. We have to show that $\mathcal{L}_{pr}(\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L))|_O = \mathfrak{f}(L)$.

Similar to Lemma 2 one can check that those $x_{K'}^{L'} \in X_{\mathfrak{f}}^{pr}$ with $L \sqsupseteq L'$ are the only arguments in $X_{\mathfrak{f}}^{pr}$ which can be $\mathtt{t}$ in a preferred labelling of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$. Now the arguments x_K^L with $K \in \mathfrak{f}(L)$ form a clique in $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$. Moreover each of these x_K^L defends itself, hence there is a preferred labelling of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$ for each $K \in \mathfrak{f}(L)$ identified by x_K^L. Let P_K be the preferred labelling with

$P_K(x_K^L) = \mathtt{t}$ where $K \in \mathfrak{f}(L)$. All $x_{K'}^{L'}$ with $K' \not\sqsubseteq K \wedge K' \not\sqsupseteq K$ are attacked by x_L^K, hence $P_K(x_{K'}^{L'}) = \mathtt{f}$. Assume $K' \sqsupset K$. If $L' \not\sqsupseteq L$, then $x_{K'}^{L'}$ is again attacked by x_L^K and $P_K(x_{K'}^{L'}) = \mathtt{f}$. If $L' \sqsupset L$, $P_K(x_{K'}^{L'}) \neq \mathtt{t}$ since it is attacked by some i or i' ($i \in I$) which is $\mathtt{u}$ in P_K. Therefore, by Lemma 3, $P_K|_O = K$.

It remains to show that there is no other preferred labelling besides these P_K with $K \in \mathfrak{f}(L)$. Towards a contradiction, assume that there is a preferred labelling P' where no x_K^L with $K \in \mathfrak{f}(L)$ is $\mathtt{t}$. By our initial considerations, those $x_{K'}^{L'}$ with $L' \sqsubset L$ are the only x-arguments which can be $\mathtt{t}$ in P'. It cannot be the case that none of them is $\mathtt{t}$, since P' would not be preferred. Then there is at least an $x_{K'}^{L'}$ which is $\mathtt{t}$ in P', with $L' \sqsubset L$, and without loss of generality we can assume that there is no $x_{K''}^{L''}$ which is $\mathtt{t}$ and $L' \sqsubset L''$. Now, since $\mathfrak{f}$ is monotonic there has to be a $K \in \mathfrak{f}(L)$ such that $K' \sqsubseteq K$. We prove that no x-argument attacking x_K^L is $\mathtt{t}$ in P'.

First, the only x-arguments that can be $\mathtt{t}$ are those $x_{K''}^{L''}$ with $L'' \sqsubset L$. Note that, according to Definition 15, $x_{K'}^{L'}$ does not attack x_K^L, since $L' \sqsubset L \wedge K' \sqsubseteq K$. If an attacker $x_{K''}^{L''}$ is attacked in turn by $x_{K'}^{L'}$ then it is $\mathtt{f}$, otherwise either $L'' \sqsubset L' \wedge K'' \sqsubseteq K'$ or $L' \sqsubset L'' \wedge K' \sqsubseteq K''$. The first case is impossible, since we would have $L'' \sqsubset L \wedge K'' \sqsubseteq K$, entailing that $x_{K''}^{L''}$ does not attack x_K^L. In the other case, by the assumption on $x_{K'}^{L'}$ it holds that $x_{K''}^{L''}$ is not $\mathtt{t}$.

Now, x_K^L defends itself against all x-arguments and none of them is $\mathtt{t}$, moreover by construction of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$, all attackers from I and I' are $\mathtt{f}$. But then, consider the labelling P'' obtained from P' by assigning to x_K^L the label $\mathtt{t}$, and by assigning to all the attackers of x_K^L the label $\mathtt{f}$. P'' is admissible and $P' \sqsubseteq P''$, contradicting the maximality of P'. $\square$

Theorem 7. *A 3-valued I/O-specification $\mathfrak{f}$ is satisfiable under pr iff $\mathfrak{f}$ is monotonic.*

As to remaining semantics, note that Propositions 7 and 8 apply to ideal and semi-stable semantics, respectively, since for each L the grounded labelling of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{gr}, L)$ coincides with the ideal labelling, and the preferred labellings of $\blacktriangleright(\mathfrak{D}_{\mathfrak{f}}^{pr}, L)$ coincide with semi-stable labellings. However, this does not allow to derive a complete characterization, since there are non-monotonic 3-valued I/O-specifications, satisfiable by ideal and semi-stable semantics, respectively.

5 Conclusions

To the best of our knowledge, this is the first characterization of the input/output expressive power of argumentation semantics. In [10], expressiveness has been studied as the capability of enforcing sets of extensions. The problem faced in this paper differs in two aspects: on the one hand, we have to enforce a set of extensions for any input rather than a single set of extensions, on the other hand we can exploit non-output arguments that are not seen outside a sub-framework. Moreover we also consider labellings besides extensions. A labelling-based investigation exploiting hidden arguments is carried out in [11], but still in the context of an ordinary AF rather than an I/O-gadget.

We restricted our considerations to total I/O-specifications, where the output is defined for each input. One can also think of situations where we do not care about the output for some inputs, i.e. the interest lies in the satisfiability of a partial function. We did not tackle these issues here, but plan to do so as part of future work.

Further future work includes the 3-valued I/O-characterization of complete semantics, being the only transparent semantics [1] for which this was left open. Moreover, the investigation of further semantics such as CF2 [4] would be of interest. Another issue is the construction of I/O-gadgets from *compact* I/O-specifications where the function is not explicitly stated but, for instance, described as a Boolean (or three-valued) circuit. We conjecture that I/O-gadgets can then be composed from simple building blocks along the lines of the given circuit. A related question in this direction is the identification of minimal I/O-gadgets satisfying a given specification.

References

1. Baroni, P., Boella, G., Cerutti, F., Giacomin, M., van der Torre, L., Villata, S.: On the Input/Output behaviour of argumentation frameworks. Artif. Intell. 217, 144–197 (2014)
2. Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. Knowledge Engineering Review 26(4), 365–410 (2011)
3. Baroni, P., Giacomin, M.: On principle-based evaluation of extension-based argumentation semantics. Artif. Intell. 171(10-15), 675–700 (2007)
4. Baroni, P., Giacomin, M., Guida, G.: SCC-Recursiveness: A general schema for argumentation semantics. Artif. Intell. 168(1-2), 162–210 (2005)
5. Caminada, M., Amgoud, L.: On the evaluation of argumentation formalisms. Artif. Intell. 171(5-6), 286–310 (2007)
6. Caminada, M., Carnielli, W.A., Dunne, P.E.: Semi-stable semantics. Journal of Logic and Computation 22(5), 1207–1254 (2012)
7. Caminada, M., Gabbay, D.M.: A logical account of formal argumentation. Studia Logica 93(2), 109–145 (2009)
8. Diller, M., Haret, A., Linsbichler, T., Rümmele, S., Woltran, S.: An extension-based approach to belief revision in abstract argumentation. In: Proc. IJCAI. pp. 2926–2932. AAAI Press (2015)
9. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. 77(2), 321–357 (1995)
10. Dunne, P.E., Dvořák, W., Linsbichler, T., Woltran, S.: Characteristics of multiple viewpoints in abstract argumentation. In: Proc. KR. pp. 72–81. AAAI Press (2014)
11. Dyrkolbotn, S.K.: How to argue for anything: Enforcing arbitrary sets of labellings using AFs. In: Proc. KR. pp. 626–629. AAAI Press (2014)
12. Modgil, S., Bench-Capon, T.J.M.: Metalevel argumentation. Journal of Logic and Computation 21(6), 959–1003 (2011)
13. Verheij, B.: Two approaches to dialectical argumentation: admissible sets and argumentation stages. In: Proc. NAIC. pp. 357–368 (1996)

Approximate Reasoning with Fuzzy-Syllogistic Systems

Bora İ Kumova

İzmir Institute of Technology; Department of Computer Engineering; 35430 Turkey
borakumova@iyte.edu.tr

Abstract. The well known Aristotelian syllogistic system consists of 256 moods. We have found earlier that 136 moods are distinct in terms of equal truth ratios that range in τ=[0,1]. The truth ratio of a particular mood is calculated by relating the number of true and false syllogistic cases the mood matches. A mood with truth ratio is a fuzzy-syllogistic mood. The introduction of (n-1) fuzzy existential quantifiers extends the system to fuzzy-syllogistic systems $^{n}\mathbb{S}$, 1<n, of which every fuzzy-syllogistic mood can be interpreted as a vague inference with a generic truth ratio that is determined by its syllogistic structure. We experimentally introduce the logic of a fuzzy-syllogistic ontology reasoner that is based on the fuzzy-syllogistic systems $^{n}\mathbb{S}$. We further introduce a new concept, the relative truth ratio $^{r}\tau$=[0,1] that is calculated based on the cardinalities of the syllogistic cases.

Keywords. Syllogistic reasoning; fuzzy logic; approximate reasoning.

1 INTRODUCTION

Multi-valued logics were initially introduced by Łukasiewicz [10], as an extension to propositional logic. After Zadeh generalised multi-valued logics within fuzzy logic [19], he discussed syllogistic reasoning with fuzzy quantifiers in the context of fuzzy logic [20]. However, this initial fuzzification of syllogistic moods was experimentally applied on only a few true moods and did not systematically cover all moods. The first systematic application of multi-valued logics on syllogisms were intermediate quantifiers and their reflection on the square of opposition [14]. However only set-theoretic representation of moods as syllogistic cases allow 64analysing the fuzzy-syllogistic systems $^{n}\mathbb{S}$ mathematically exactly, such as by calculating truth ratios of moods [6] and their algorithmic usage in fuzzy inferencing [7]. Here we present a sample application of $^{n}\mathbb{S}$ for fuzzy-syllogistic ontology reasoning.

Learning from scratch can be modelled probabilistically, as objects and their relationships need to be first synthesised from a statistically significant number of perceived instances of similar objects. This leads to probabilistic ontologies [4], [15], [11], in which attributes of objects may be synthesised also as objects.

There are more probabilistic ontology reasoners than fuzzy or possibilistic ones and most of them reason with probabilist ontologies [8]. Several ontology reasoners employ possibilistic logic and reason with fuzzy ontologies. The most popular reasoning logic being hyper-tableau, for instance in HermiT [12]. Other experimental reasoning logics are also interesting to analyse, such as fuzzy rough sets and Łukasiewicz logic [3] in FuzzyDL [1], Zadeh and Gödel fuzzy operators in DeLorean [2], Mamdani inference in HyFOM [18] or possibilistic logic in KAON [15]. Fuzzy-syllogistic reasoning (FSR) can be seen as a generalisation of both, fuzzy-logical and possibilistic reasoners.

A fuzzy-syllogistic ontology (FSO) extends the concept of ontology with the quantities that led to the ontological concepts. A FSO is usually generated probabilistically, but does not preserve any probabilities like probabilistic ontologies [11] or probabilistic logic networks [5] do. A FSO can be a fully connected and bidirectional graph.

Several generic reasoning logics are discussed in the literature, like probabilistic, non-monotonic or non-axiomatic reasoning [17]. Fuzzy-syllogistic reasoning in its basic form [21] is possibilistic, monotonic and axiomatic.

Syllogistic reasoning reduced to the proportional inference rules deduction, induction and abduction are employed in the Non-Axiomatic Reasoning System (NARS) [16]. Whereas FSR uses the original syllogistic moods and their fuzzified extensions [22].

There is one implementation mentioned in the literature that is close to the concept of syllogistic cases: Syllogistic Epistemic REAsoner (SEREA) implements poly-syllogisms and generalised quantifiers that are associated with combinations of distinct spaces, which are mapped onto some interval arithmetic. Reasoning is then performed with concrete quantities, determined with the interval arithmetic [13].

First the fuzzy-syllogistic systems ${}^{n}\mathbb{S}$ are discussed, thereafter fuzzy-syllogistic reasoning is introduced, followed by its sample application on a fuzzy-syllogistic ontology and the introduction of relative truth ratios ${}^{r}\tau$.

2 FUZZY-SYLLOGISTIC SYSTEMS

The fuzzy-syllogistic systems ${}^{n}\mathbb{S}$, with $1<n$ fuzzy quantifiers, extend the well known Aristotelian syllogisms with fuzzy-logical concepts, like truth ratio for every mood and fuzzy quantifiers or in general fuzzy sets. We discuss first the systems ${}^{n}\mathbb{S}$ and introduce them further below as the basic reasoning logic of FSR.

2.1 Aristotelian Syllogistic System $\mathbb{S}$

The Aristotelian syllogistic system $\mathbb{S}$ consists of inclusive existential quantifiers ψ, ie I includes A and O includes E as one possible case:

Universal affirmative: All S are P: ψ=A: $\{x \mid x \notin P\text{-}S \wedge x \in P \cap S\}$

Universal negative: All S are not P: ψ=E: $\{x \mid x \in S\text{-}P \wedge x \notin P\text{-}S\}$

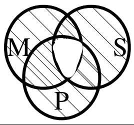
Inclusive existential affirmative: Some S are P: ψ=I: $A \cup \{x \mid (x \notin S\text{-}P \wedge x \notin P\text{-}S \wedge x \in P \cap S) \vee (x \notin S\text{-}P \wedge x \in P \cap S)\}$

Inclusive existential negative: Some S are not P: ψ=O: $E \cup \{x \mid (x \in S\text{-}P \wedge x \notin P\text{-}S \wedge x \notin P \cap S) \vee (x \in S\text{-}P \wedge x \notin P \cap S)\}$

A categorical syllogism $\psi_1\psi_2\psi_3F$ is an inference schema that concludes a quantified proposition $\Phi_3=S\psi_3P$ from the transitive relationship of two given quantified proportions $\Phi_1=\{M\psi_1P, P\psi_1M\}$ and $\Phi_2=\{S\psi_2M, M\psi_2S\}$:

$$\psi_1\psi_2\psi_3F = (\Phi_1=\{M\psi_1P, P\psi_1M\}, \Phi_2=\{S\psi_2M, M\psi_2S\}, \Phi_3=S\psi_3P)$$

Table 1: Binary coding of the 7 possible distinct spaces for three sets.

Syllogistic Case Δ_{95}		
Binary code $\Delta_j = \delta_1\delta_2\delta_3\delta_4\delta_5\delta_6\delta_7$ *	Venn Diagram	Space Diagram+
Δ_{96}=1111110#		 δ_1 δ_2 δ_3 δ_4 δ_5 δ_6 δ_7

*Binary coding of all possible distinct space combinations Δ_j, j=[1,96] that can be generated for three sets.
#δ_i=0: space i is empty; δ_i=1: space i is not empty; i=[1,7].
+Every circle of a space diagram represents exactly one distinct sub-set of M∪ P∪ S.

where F={1, 2, 3, 4} identifies the four possible combinations of Φ_1 with Φ_2, namely syllogistic figures. Every figure produces 4^3=64 moods and the whole syllogistic system 𝕊 has 4x64=256 moods.

2.2 Syllogistic-Cases

Syllogistic cases are an elementary concept of the fuzzy-syllogistic systems $^n\mathbb{S}$, for calculating truth ratios [6] of the moods algorithmically [7].

For three sets, 7 distinct spaces δ_i, i=[1,7] are possible, which can be easily identified in a Venn diagram (Table 1). There are in total j=96 distinct combinations of the spaces $\Delta_j=\delta_1\delta_2\delta_3\delta_4\delta_5\delta_6\delta_7$, j=[1,96] [22], which constitute the universal set of syllogistic moods. Out of this universe, we determine for every mood true and false matching space combinations (Fig 1).

2.3 Fuzzy-Syllogistic Moods

We extend the ancient binary truth classification of moods, to a fuzzy classification with truth values in [0,1]. For this purpose, first the above set-theoretical definitions of the quantifiers of a particular mood are compared against the set of all syllogistic cases Δ_j, j=[1,96], in order to identify true and false matching cases:

$$\textbf{True syllogistic cases: } \Lambda^t = {}_{j=1}\cup^{96}\Delta_j \in (\Phi^\Delta{}_1 \cap \Phi^\Delta{}_2) \rightarrow \Delta_j \in \Phi^\Delta{}_3$$

$$\textbf{False syllogistic cases: } \Lambda^f = {}_{j=1}\cup^{96}\Delta_j \in (\Phi^\Delta{}_1 \cap \Phi^\Delta{}_2) \rightarrow \Delta_j \notin \Phi^\Delta{}_3$$

where Λ^t and Λ^f is the set of all true and false matching cases of a particular mood, respectively (Fig 1) and Φ^Δ is a proposition in terms of syllogistic cases. For instance, the two premisses Φ_1 and Φ_2 of the mood IAI4 of the syllogistic system 𝕊, match the 10 syllogistic cases $\Lambda^t = \{\Delta_4, \Delta_{19}, \Delta_{67}, \Delta_{24}, \Delta_{43}, \Delta_{46}, \Delta_{68}, \Delta_{74}, \Delta_{48}, \Delta_{76}\}$, which are all true for the conclusion Φ_3 as well. Thus the mood has no false cases $\Lambda^f = \emptyset$.

The truth ratio of a mood is then calculated by relating the amounts of the two sets Λ^t and Λ^f with each other. Consequently the truth ratio becomes either more true or more false $\tau \in \{\tau^f, \tau^t\}$:

$$\textbf{More true: } \tau^t \in \{|\Lambda^f|<|\Lambda^t| \rightarrow 1-|\Lambda^f|/(|\Lambda^t|+|\Lambda^f|)\} = [0.545,1]$$

$$\textbf{More false: } \tau^f \in \{|\Lambda^t|<|\Lambda^f| \rightarrow |\Lambda^t|/(|\Lambda^t|+|\Lambda^f|)\} = [0,0.454]$$

where $|\Lambda^t|$ and $|\Lambda^f|$ are the numbers of true and false syllogistic cases, respectively. A fuzzy-syllogistic mood is then defined by assigning an Aristotelian mood $\psi_1\psi_2\psi_3F$ the structurally fixed truth ratio τ:

Fuzzy-syllogistic mood: $(\psi_1\psi_2\psi_3F, \tau)$

The truth ratio identifies the degree of truth of a particular mood, which we will associate further below in fuzzy-syllogistic reasoning with generic vagueness of inferencing with that mood.

The analysis of the Aristotelian syllogistic system $\mathbb{S}$ with these concepts reveals several interesting properties, like $\mathbb{S}$ has 136 distinct moods, 25 true moods τ=1, of which 11 are distinct, and 25 false moods τ=0, of which 11 are distinct, and that $\mathbb{S}$ is almost point-symmetric on syllogistic cases and truth ratios of the moods [22], [9].

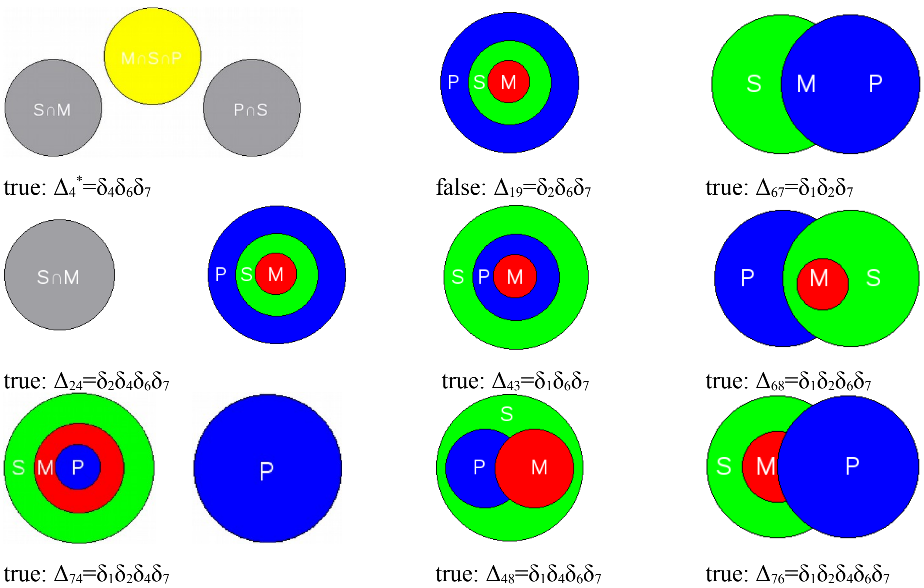
2.4 Fuzzy-Syllogistic System $^2\mathbb{S}$

In the fuzzy-syllogistic system (FSS) $^2\mathbb{S}$, the universal cases A and E are excluded from the existential quantifiers I and O, respectively:

Exclusive existential affirmative: Some S butNotAll are P: ψ=I: $\{x|\ (x\notin S\text{-}P \wedge x\notin P\text{-}S \wedge x\in P\cap S) \vee (x\notin S\text{-}P \wedge x\in P\cap S)\}$

Exclusive existential negative: Some S butNotAll are not P: ψ=O: $\{x|\ (x\in S\text{-}P \wedge x\notin P\text{-}S \wedge x\notin P\cap S) \vee (x\in S\text{-}P \wedge x\notin P\cap S)\}$

For instance the mood IAI4 of $\mathbb{S}$, becomes $^{2/1}IA^1I4$ in $^2\mathbb{S}$. Because of the exclusive existential quantifier $^{2/1}I$, the case Δ_{46} is no more matched by of the first premiss Φ_1 and the conclusion Φ_3 becomes false for the case Δ_{19} (Fig 1).



* A full list of all syllogistic cases Δ_j, j=[1,96], can be found elsewhere [22].

Fig 1. 9 syllogistic cases Δ_j of the mood $^{2/1}IA^1I4$ of the fuzzy-syllogistic systems $^2\mathbb{S}$.

Table 2. Value ranges of affirmative quantifiers of various fuzzy-syllogistic systems $^{n}\mathbb{S}$

Syllogistic System		Quantifier ψ*					
Aristotelian	$\mathbb{S}$	A=all	I=some (including A)				
Fuzzy	$^{2}\mathbb{S}$	A=all	$^{2/1}$I=some (excluding A)				
	$^{3}\mathbb{S}$	A=all	$^{3/2}$I=most	$^{3/1}$I=several			
	$^{4}\mathbb{S}$	A=all	$^{4/3}$I=most	$^{4/2}$I=half	$^{4/1}$I=several		
	$^{5}\mathbb{S}$	A=all	$^{5/4}$I=many	$^{5/3}$I=most	$^{5/2}$I=several	$^{5/1}$I=few	
	$^{6}\mathbb{S}$	A=all	$^{6/5}$I=many	$^{6/4}$I=most	$^{6/3}$I=half	$^{6/2}$I=several	$^{6/1}$I=few
	$^{n}\mathbb{S}$	A=all	$^{n/n-1}$I	...			$^{n/1}$I

* Column breadths are not drawn proportional to the overall value range and to the other quantifiers.

The analysis of the FSS $^{2}\mathbb{S}$ shows that $^{2}\mathbb{S}$ has 70 distinct moods, 11 true moods τ=1, of which 5 are distinct, and 40 false moods τ=0, of which 13 are distinct, and that $^{2}\mathbb{S}$ is not point-symmetric [22], [9].

2.5 Fuzzy-Syllogistic System $^{n}\mathbb{S}$

By using (n-1) fuzzy-existential quantifiers, the total number of fuzzy-syllogistic moods of the FSS $^{n}\mathbb{S}$ increases to $(2n)^3$. The sample mood IAI4 of $\mathbb{S}$ can now be generalised to $^{n/k1}$IA^{k2}I4, $1<n$, $0<k_1,k_2<n$ of $^{n}\mathbb{S}$. $^{n/k1}$IA^{k2}I4 consists of $(n-1)^2$ fuzzy-moods, all having the very same 9 syllogistic cases (Fig 1).

Same linguistic terms used in different FSSs do not necessarily equal each other. For instance, "most" may have different value ranges in the FSSs $^{3}\mathbb{S}$, $^{4}\mathbb{S}$, $^{5}\mathbb{S}$, $^{6}\mathbb{S}$ and therefore are in general not equal $^{3/2}I\neq{}^{4/3}I\neq{}^{5/3}I\neq{}^{6/4}I$, respectively. Likewise for "half" in $^{4}\mathbb{S}$ and $^{6}\mathbb{S}$ the quantifiers may not exactly equal $^{4/2}I\neq{}^{6/3}I$, respectively (Table 2).

3 FUZZY-SYLLOGISTIC ONTOLOGY

A fuzzy-syllogistic ontology (FSO) consists of concepts, their relationships and assertions on them, whereby all quantities are given with fuzzy-quantifications:

Fuzzy-syllogistic ontology: FSO=^{k}(C, R, A)

where C is the set of all concepts, R is the set of all directed relationships between the concepts and A is the set of all assertions. A FSO may be specified top-down or may be transformed from any existing ontology, provided that all quantities are determined systematically, in compliance with one of the FSSs $^{k}\mathbb{S}$, $1<k\leq n$, (Table 2). In a bottom-up approach, a FSO may be learned from given domain data.

3.1 Learning Fuzzy Quantifiers

Although existing learning approaches generate ontological concepts and their relationships through probabilistic analysis of the data [4], [15], [11], the quantities that actually imply the concepts and relationships, are not preserved in the ontology [8]. Therefore we sketch here briefly how to learn such quantities of a FSO.

For any directly connected triple concept relationship of the FSO, seven distinct relationships are possible (Table 1). The quantity of every such relationship has to be stored with the FSO. Since the relationships may be bi-directional or a concept may be involved in multiple triple relationships (Fig 2), the quantities of all these cases need to be stored too.

The objective of learning a FSO=^{k}(C, R, A) is, to update the FSO against changing domain data and to determined the most appropriate FSS ${}^{k}\mathbb{S}$ out of ${}^{n}\mathbb{S}$, $1<k\leq n$.

3.2 Relative truth ratio

Relative truth ratios are calculated from the exact quantities of all syllogistic cases of a particular mood, rather than from just the amount of the cases:

$$\textbf{Relative true: } {}^{r}\tau^{t} = \lambda^{f}<\lambda^{t} \rightarrow \lambda^{f}/(\lambda^{f}+\lambda^{t})$$

$$\textbf{Relative false: } {}^{r}\tau^{f} = \lambda^{t}<\lambda^{f} \rightarrow \lambda^{t}/(\lambda^{t}+\lambda^{f})$$

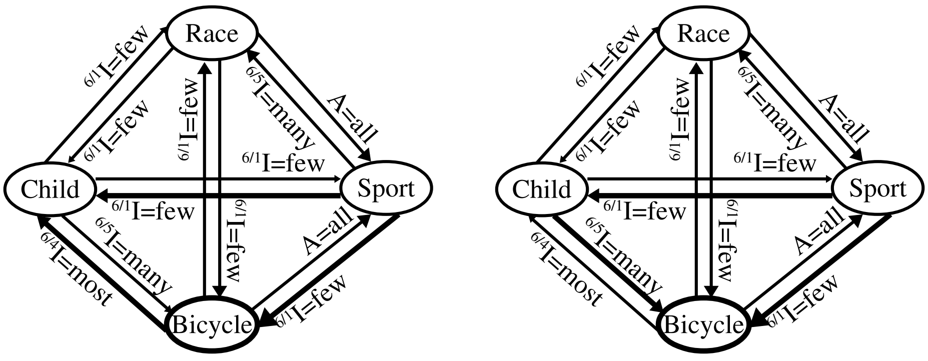
where $\lambda^{t} = {}_{j=1}\sum^{|\Lambda t|} |\Delta^{t}_{j}|$ and $\lambda^{f} = {}_{j=1}\sum^{|\Lambda f|} |\Delta^{f}_{j}|$ is the total number of elements accumulated over all true and false syllogistic cases, respectively. Where $|\Lambda^{t}|$ and $|\Lambda^{f}|$ is the number of true and false cases of the mood, respectively. Accordingly, we can re-define a fuzzy-syllogistic mood with relative truth ratio ${}^{r}\tau$:

$$\textbf{Fuzzy-syllogistic mood with relative truth ratio: } (\psi_1\psi_2\psi_3 F, {}^{r}\tau)$$

The structural truth ratio τ of a particular mood represents the generic vagueness of the mood and is constant, whereas the relative truth ratio ${}^{r}\tau$ adjusts τ by weighting every case of the mood with its actual quantity.

4 FUZZY-SYLLOGISTIC REASONING

The fuzzy-syllogistic systems $\mathbb{S}$, ${}^{2}\mathbb{S}$ and ${}^{6}\mathbb{S}$ are currently implemented experimentally as the reasoning logic of the fuzzy-syllogistic reasoner (FSR), for reasoning over FSOs [21]. Our objective is to generalise the logic of the reasoner to ${}^{n}\mathbb{S}$ and to use it as a cognitive primitive for modelling other cognitive concepts within a cognitive architecture. We now sketch the algorithmic design of the FSR.



${}^{6}\mathbb{S}$: (${}^{6/4}$I^{1}I^{1}I1, 40/47=0.851)
Φ_1: Most **bicycles** are good for children
Φ_2: Few sports are good for **bicycles**
Φ_3: Few **sports** are good for **children**

${}^{6}\mathbb{S}$: (${}^{6/5}$I^{1}I^{1}I2, 40/48=0.833)
Φ_1: Many children have **bicycles**
Φ_2: Few sports are good for **bicycles**
Φ_3: Few **sports** are good for **children**

Fig 2. Sample fuzzy-syllogistic ontology with affirmative relationships and the best matching fuzzy-syllogistic moods from the syllogistic figures 1 and 2.

4.1 Reasoning Algorithm

FSR is concerned with identifying for any given concept c∈C, all possible triple concept relationships r∈R, r={M,P,S}, of the given FSO=^{k}(C, R, A) and to reason with the most appropriate fuzzy-syllogistic moods of its FSS $^k\mathbb{S}$. Whereby associated assertions a∈A may be used for exemplifying a particular reasoning.

For instance, for the concept c=Bicycle, multiple triple relationships r={Bicycle, Child, Sports} exist in the sample FSO=^{6}(C, R, A) (Fig 2). The reasoner iterates for the FSSs $^k\mathbb{S}$, k=[2,n], and for their moods, in order to match the moods with the closest fuzzy-syllogistic quantities of relationships r. The reasoner determines the FSS k=6 and the mood $^{6/k1}$IA^{k2}I4, 0<k_1,k_2<6 as best matches for this example.

In the below example with $\mathbb{S}$, I in Φ_3 may include A and therefore is less true. Whereas in $^3\mathbb{S}$, $^{3/1}$I in Φ_3 is still too general. The best matching quantifiers are found in $^6\mathbb{S}$ (Fig 3).

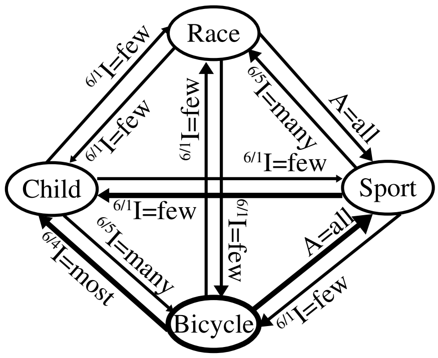
$\mathbb{S}$: (IAI3, 10/10=1.0)
Φ_1: Some **bicycles** are good for children
Φ_2: All **bicycles** are good for sports
Φ_3: Some **sports** are good for **children**

$^3\mathbb{S}$: ($^{3/2}$IA^{1}I3, 6/6=1.0)
Φ_1: Most **bicycles** are good for children
Φ_2: All **bicycles** are good for sports
Φ_3: Several **sports** are good for **children**

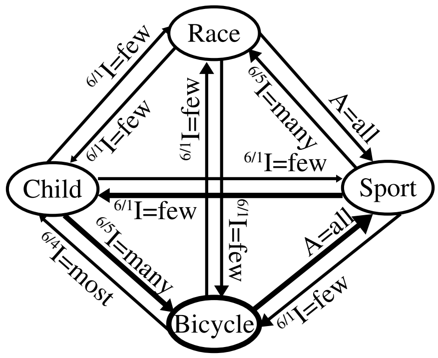
5 CONCLUSION

The FSSs $\mathbb{S}$, $^2\mathbb{S}$, $^6\mathbb{S}$ were introduced as the fundamental logic of the fuzzy-syllogistic reasoner (FSR) and its usage was exemplified on a sample fuzzy-syllogistic ontology (FSO). The relative truth ratio $^r\tau$ of a mood was introduced, which adapts the structural truth ratio τ of the mood to the amount of elements of its syllogistic cases. FSR with FSOs is a generic possibilistic reasoning approach, since the employed reasoning logic $^n\mathbb{S}$ is generic.

We are currently implementing a sample educational application that extends an existing probabilist ontology learning tool and generates a FSO=^{k}(C, R, A) for a given



$^6\mathbb{S}$: ($^{6/4}$IA^{1}I3, 6/6=1.0)
Φ_1: Most **bicycles** are good for children
Φ_2: All **bicycles** are good for sports
Φ_3: Few **sports** are good for **children**



$^6\mathbb{S}$: ($^{6/5}$IA^{1}I4, 8/9=0.888)
Φ_1: Many children have **bicycles**
Φ_2: All **bicycles** are good for sports
Φ_3: Few **sports** are good for **children**

Fig 3. Sample fuzzy-syllogistic ontology with affirmative relationships and the best matching fuzzy-syllogistic moods from the syllogistic figures 3 and 4.

domain. For a user-chosen concept C from the ontology FSO, FSR is then used to reason with all associated quantities R and present the user all associated scenarios A.

6 REFERENCES

[1] Bobillo, F; Straccia, U; 2008; "fuzzyDL: an expressive fuzzy description logic reasoner"; Fuzzy Systems (FUZZ-IEEE); IEEE Computer Society
[2] Bobillo, F; Delgado, M; Gomez-Romero, J; 2008; "DeLorean: a reasoner for fuzzy OWL 1.1"; Uncertainty Reasoning for the Semantic Web (URSW); Springer LNAI
[3] Bobillo, F; Straccia, U; 2011; "Reasoning with the finitely many-valued Łukasiewicz fuzzy Description Logic SROIQ"; Information Sciences, 181
[4] Cimiano, P; Völker, J; 2005; "Text2Onto: A Framework for Ontology Learning and Data-driven Change Discovery"; International conference on Natural Language Processing and Information Systems (NLDB); Springer
[5] Goertzel, B; Iklé, M; Goertzel, ILF; Heljakka, A; 2008; "Probabilistic Logic Networks: A Comprehensive Conceptual, Mathematical and Computational Framework for Uncertain Inference"; Springer
[6] Kumova, Bİ; Çakır, H; 2010; "The Fuzzy Syllogistic System"; Mexican International Conference on Artificial Intelligence (MICAI); LNAI; Springer
[7] Kumova, Bİ; Çakır, H; 2010; "Algorithmic Decision of Syllogisms"; International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA-AIE); Córdoba; LNCS; Springer
[8] Kumova, Bİ; 2015; "Generating Ontologies from Relational Data with Fuzzy-Syllogistic Reasoning"; Beyond Databases, Architectures and Structures (BDAS); Springer, Communications in Computer and Information Science (CCIS)
[9] Kumova, Bİ; 2015; "Properties of the Syllogistic System"; in review
[10] Łukasiewicz, J; 1920; "O logice trójwartościowej" (translation from Polish: On three-valued logic); Ruch filozoficzny, vol 5
[11] Lukasiewicz, T; Straccia, U; 2008; "Managing uncertainty and vagueness in description logics for the SemanticWeb"; Web Semantics: Science, Services, Agents on WWW; Elsevier
[12] Motik, B; Shearer, R; Horrocks, I; 2009; "Hypertableau Reasoning for Description Logics"; Journal of Artificial Intelligence Research, 36; AAAI
[13] Pereira-Fariña, M; Vidal, JC; Díaz-Hermida, F; Bugarín, A; 2014; "A fuzzy syllogistic reasoning schema for generalized quantifiers"; Fuzzy Sets and Systems, vol 234
[14] Peterson, P; 2000; "Intermediate Quantifiers: Logic, Linguistics, and Aristotelian Semantics"; Ashgate
[15] Qi, G; Pan, JZ; Ji, Q; 2007; "A possibilistic extension of description logics"; Description Logics (DL); Sun SITE Central Europe (CEUR)
[16] Wang, P; 1994; "From Inheritance Relation to Non-Axiomatic Logic"; International Journal of Approximate Reasoning, vol 7
[17] Wang, P; 1995; "Reference Classes and Multiple Inheritances"; International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, vol 3/1
[18] Yaguinuma, CA; Magalhães Jr, WCP; Santos, MTP; Camargo, HA; Reformat, M; 2014; "Combining Fuzzy Ontology Reasoning and Mamdani Fuzzy Inference System with HyFOM Reasoner"; Enterprise Information Systems; LNBI, 190
[19] Zadeh, LA; 1975; "Fuzzy Logic and Approximate Reasoning"; Syntheses 30
[20] Zadeh, LA; 1985; "Syllogistic reasoning in fuzzy logic and its application to usuality and reasoning with dispositions"; IEEE Transactions on Systems, Man and Cybernetics, 15/6
[21] Zarechnev, M; Kumova, Bİ; 2015; "Ontology-Based Fuzzy-Syllogistic Reasoning"; Industrial, Engineering & Applications of Applied Intelligent Systems (IEA-AIE); LNCS
[22] Zarechnev, M; Kumova, Bİ; 2015; "Truth Ratios of Syllogistic Moods"; IEEE International Conference on Fuzzy Systems (FUZZ-IEEE); IEEE Xplore