technische universität
dortmund

department of
computer science

TECHNICAL REPORTS IN COMPUTER SCIENCE

Technische Universität Dortmund

Proceedings of the

16th International Workshop on Non-Monotonic Reasoning

(NMR 2016)

April 22 – 24, 2016

Cape Town, South Africa

Editors:
Gabriele Kern-Isberner, Lehrstuhl Informatik 1, Technische Universität Dortmund
Renata Wassermann, Computer Science Department, University of São Paulo

Number: 852

Technische Universität Dortmund — Fakultät für Informatik
Otto-Hahn-Str. 14, 44227 Dortmund

Gabriele Kern-Isberner, Renata Wassermann (Editors), *Proceedings of the 16th International Workshop on Non-Monotonic Reasoning (NMR 2016)*, Cape Town, South Africa; April 22 – 24, 2016. ©2016.

Nonmonotonicity is crucial for any formal approach to modelling human reasoning – most of the conclusions we draw in our everyday lives and on which we base (sometimes important) decisions are defeasible, prone to be given up if further information arrives. This is in clear contrast to classical – e.g., propositional or first-order – logics which are monotonic, i.e., their deductive conclusions are preserved for eternity. It is due to monotonicity that all (correct) proofs in mathematics are still valid, regardless of whatever new theories are being developed, but also that robots based on classical logics fail in uncertain, incompletely specified environments.

Therefore, nonmonotonic reasoning (NMR) deals with important issues in Artificial Intelligence, and has strong connections to other areas of knowledge representation, in particular, to belief revision, action logics, argumentation, logic programming, preference handling, and uncertain reasoning. The NMR workshops are the premier forum for presenting results in this broad subfield of knowledge representation and reasoning (KR). Their aim is to bring together active researchers, and foster discussions and collaborations on theoretical foundations, applications, and system development.

NMR has a long history – it started in 1984, and is held every two years. Recent previous NMR workshops took place in Vienna (2014), Rome (2012), Toronto (2010), and Sydney (2008). Following established and fruitful traditions, NMR 2016 was co-located with the *15th International Conference on Principles of Knowledge Representation and Reasoning (KR 2016)* and the *29th International Workshop on Description Logics (DL 2016)*. In particular, NMR 2016 shared a joint session with DL 2016. We were happy to welcome *Laura Giordano* (Universitá del Piemonte Orientale) and *Leon van der Torre* (University of Luxembourg) as invited speakers, Laura's talk was also invited by DL 2016.

This volume contains most of the accepted papers of NMR 2016. Some papers had been already published, or are meant to be published elsewhere, so we could only provide URL's in those cases. This collection of NMR papers illustrate impressively both the depth and the breadth of NMR by dealing with theoretical issues as well as connecting different subfields of knowledge representation and reasoning.

In *Studies on Brutal Contraction and Severe Withdrawal: Preliminary Report*, Marco Garapa, Eduardo Fermé, and Maurício Reis study different classes of contraction operators and provide axiomatic characterizations for them. Zhiqiang Zhuang, James Delgrande, Abhaya Nayak, and Abdul Sattar deal with interleaving two kinds of nonmonotonicity: in their paper *A New Approach for Revising Logic Programs*, they also allow the logic underlying belief revision operations to be nonmonotonic. Also Aaron Hunter's paper *Ordinal Conditional Functions for Nearly Counterfactual Revision* considers an interesting and often neglected issue in belief revision: How can revision by (nearly) counterfactual conditionals be carried out? Marc van Zee and Dragan Doder present results on *AGM-Style Revision of Beliefs and Intentions from a Database Perspective*.

Two papers study belief revision in a probabilistic environment: Gavin Rens proposes a unified model of quantitative belief change in his paper *On Stochastic Belief Revision and Update and their Combination*, and together with Thomas Meyer and Giovanni Casini, he also contributes to this volume by *Revising Incompletely Specified Convex Probabilistic Belief Bases*. Moreover, another two papers deal with connections between nonmonotonic reasoning and belief revision, on the one hand, and ontological reasoning, on the other hand: Özgür Özçep considers *Iterated Ontology Revision by Reinterpretation*, and Valentina Gliozzi focusses on typicality operators in her paper *A strengthening of rational closure in DLs: reasoning about multiple aspects*.

One of the most basic ideas of nonmonotonic reasoning is to order models or states by preference relations. In *Preferential Modalities Revisited*, Katarina Britz and Ivan Varzinczak apply such semantic preference relations to modal accessibility relations, while Kristijonas Čyras and Francesca Toni consider preferences in assumption-based argumentation (ABA) frameworks in their paper *Properties of ABA$^+$ for Non-Monotonic Reasoning*. Jesse Heyninck and Christian Straßer also emphasize the strong *Relations between assumption-based approaches in nonmonotonic logic and formal argumentation*. Ringo Baumann, Thomas Linsbichler, and Stefan Woltran deal with *Verifiability of Argumentation Semantics* by elaborating on which specific information some well-known semantics of abstract argumentation systems can be based. Jean-Guy Mailly's paper *Using Enthymemes to Fill the Gap between Logical Argumentation and Revision of Abstract Argumentation Frameworks* is a first approach to tackle the problem that agents cannot decode arguments correctly due to missing or different (background) knowledge. Thomas Linsbichler, Jörg Pührer, and Hannes Strass aim at *Characterizing Realizability in Abstract Argumentation* by presenting algorithmic approaches that are apt to build up knowledge bases of arguments from a given set of interpretations. A somehow dual problem is considered in the context of belief merging in the paper *Distributing Knowledge into Simple Bases* by Adrian Haret, Jean-Guy Mailly, and Stefan Woltran: How can a knowledge base arise by merging simpler knowledge bases in a given fragment of classical logic?

Zeynep Saribatur and Thomas Eiter present a high-level representation formalism that can be applied to model *Reactive Policies with Planning for Action Languages*. In his extended abstract, Zoltán Ésik studies *Equational properties of stratified least fixed points* associated with logic programs. Adrian Paschke and Tara Athan show in their paper *Law Test Suites for Semantically-Safe Rule Interchange* how basic principles of nonmonotonic reasoning can be adapted to verify a particular semantics. In *Static and Dynamic Views on the Algebra of Modular Systems*, Eugenia Ternovska elaborates on properties of a knowledge representation framework called Algebra of Modular Systems; in particular, she uses the algebra for a high-level encoding of problem solving on graphs.

ACKNOWLEDGMENTS

First of all, we would like to thank the members of our Program Committee who provided us with excellent reviews on time. We are also grateful to the South African Department of Science and Technology (DST), the Council for Scientific and Industrial Research (CSIR) in South Africa, The South African Centre for Artificial Intelligence Research (CAIR), Principles of Knowledge Representation and Reasoning, Incorporated (KR, Inc.), and the European Association for Artifical Intelligence (EurAI, formerly ECCAI) for supporting our workshop by donations. The team of local organizers – Tommie Meyer in the first place – did a perfect job to help us making NMR 2016 a great event, thanks to all of them.

Last, but not least, we wish to thank Christian Eichhorn who took care of the NMR 2016 website and set up these proceedings, and the Technische Universität Dortmund where the proceedings were published as a technical report of the Faculty of Computer Science.

PROGRAM COMMITTEE OF NMR 2016

Gabriele Kern-Isberner, Technische Universität Dortmund (Co-Chair)
Renata Wassermann, University of São Paulo (Co-Chair)
Christoph Beierle, University of Hagen
Alexander Bochman, Holon Institute of Technology
Gerhard Brewka, Leipzig University
Jan Broersen, Utrecht University
Marina De Vos, University of Bath
Marc Denecker, K.U.Leuven
Juergen Dix, Clausthal University of Technology
Paul Dunne, University of Liverpool
Wolfgang Faber, University of Huddersfield
Eduardo Fermé, Universidade da Madeira
Martin Gebser, University of Potsdam
Michael Gelfond, Texas Tech University
Valentina Gliozzi, Universitá di Torino
Lluis Godo, Artificial Intelligence Research Institute, Barcelona
Sven Ove Hansson, Royal Institute of Technology, Stockholm
Andreas Herzig, Université Paul Sabatier, Toulouse
Anthony Hunter, University College London
Katsumi Inoue, National Institute of Informatics, Japan
Tomi Janhunen, Aalto University
Sébastien Konieczny, Université d'Artois
Gerhard Lakemeyer, Aachen University of Technology
Thomas Lukasiewicz, University of Oxford
Maria Vanina Martinez, Universidad Nacional del Sur (Bahia Blanca)
Thomas Meyer, University of Cape Town

CONTENTS

Invited Talks

Regular Papers

Reasoning about typicality in preferential description logics

Laura Giordano
Dipartimento di Scienze e Innovazione Tecnologica
Istituto di Informatica
Università del Piemonte Orientale
laura.giordano@uniupo.it

Abstract

The talk presents an approach for reasoning about typicality in description logics (DLs), based on Kraus, Lehmann and Magidor's preferential semantics. It describes an extension to DLs of Lehmann and Magidor's rational closure as well as a semantic characterization for it through a minimal model semantics. For expressive description logics, the computation of rational closure can exploit a polynomial encoding of preferential entailment into standard DLs. The talk aims at discussing to what extent preferential reasoning can be considered as a building block for dealing with exceptions in OWL ontologies, considering that dealing with exceptions is a long standing problem of nonmonotonic reasoning, and at comparing this approach with other approaches to defeasible reasoning in DLs.

Arguing about obligations and permissions

Leon van der Torre
University of Luxembourg

Abstract

In this talk I apply a theory of structured argumentation to normative reasoning. In an ASPIC+ style setting, I discuss the definition of argument and attack, the role of constitutive and permissive norms, and hierarchical normative systems. Based on joint work with Beishui Liao, Nir Oren, Gabriella Pigozzi and Serena Villata.

Verifiability of Argumentation Semantics[*]

Ringo Baumann
Leipzig University
Germany

Thomas Linsbichler and **Stefan Woltran**
TU Wien
Austria

Abstract

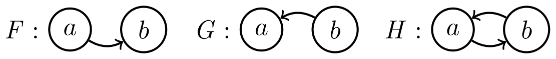
Dung's abstract argumentation theory is a widely used formalism to model conflicting information and to draw conclusions in such situations. Hereby, the knowledge is represented by so-called argumentation frameworks (AFs) and the reasoning is done via semantics extracting acceptable sets. All reasonable semantics are based on the notion of conflict-freeness which means that arguments are only jointly acceptable when they are not linked within the AF. In this paper, we study the question which information on top of conflict-free sets is needed to compute extensions of a semantics at hand. We introduce a hierarchy of so-called verification classes specifying the required amount of information. We show that well-known standard semantics are exactly verifiable through a certain such class. Our framework also gives a means to study semantics lying inbetween known semantics, thus contributing to a more abstract understanding of the different features argumentation semantics offer.

Introduction

In the late 1980s the idea of using *argumentation* to model nonmonotonic reasoning emerged (see (Loui 1987; Pollock 1987) as well as (Prakken and Vreeswijk 2002) for excellent overviews). Nowadays argumentation theory is a vibrant subfield of Artificial Intelligence, covering aspects of knowledge representation, multi-agent systems, and also philosophical questions. Among other approaches which have been proposed for capturing representative patterns of inference in argumentation theory (Besnard et al. 2014), Dung's abstract argumentation frameworks (AFs) (Dung 1995) play an important role within this research area. At the heart of Dung's approach lie the so-called *argumentation semantics* (cf. (Baroni, Caminada, and Giacomin 2011) for an excellent overview). Given an AF F, which is set-theoretically just a directed graph encoding arguments and attacks between them, a certain argumentation semantics σ returns acceptable sets of arguments $\sigma(F)$, so-called *σ-extensions*. Each of these sets represents a reasonable position w.r.t. F and σ.

Over the last 20 years a series of abstract argumentation semantics were introduced. The motivations of these semantics range from the desired treatment of specific examples to fulfilling a number of abstract principles. The comparison via abstract criteria of the different semantics available is a topic which emerged quite recently in the community ((Baroni and Giacomin 2007b) can be seen as the first paper in this line). Our work takes a further step towards a comprehensive understanding of argumentation semantics. In particular, we study the following question: Do we really need the entire AF F to compute a certain argumentation semantics σ? In other words, is it possible to unambiguously determine acceptable sets w.r.t. σ, given only partial information of the underlying framework F. In order to solve this problem let us start with the following reflections:

1. As a matter of fact, one basic requirement of almost all existing semantics[1] is that of conflict-freeness, i.e. arguments within a reasonable position are not allowed to attack each other. Consequently, knowledge about conflict-free sets is an essential part for computing semantics.
2. The second step is to ask the following: Which information on top on conflict-free sets has to be added? Imagine the set of conflict-free sets given by $\{\emptyset, \{a\}, \{b\}\}$. Consequently, there has to be at least one attack between a and b. Unfortunately, this information is not sufficient to compute any standard semantics (except naive extensions, which are defined as $\subseteq$-maximal conflict-free sets) since we know nothing precise about the neighborhood of a and b. The following three AFs possess exactly the mentioned conflict-free sets, but differ with respect to other



3. The final step is to try to minimize the added information. That is, which kind of knowledge about the neighborhood is somehow dispensable in the light of computation? Clearly, this will depend on the considered semantics. For instance, in case of stage semantics (Verheij 1996), which requests conflict-free sets of maximal range, we do not need any information about incoming attacks. This information can not be omitted in case of admissible-based semantics since incoming attacks require counterattacks.

The above considerations motivate the introduction of so-called verification classes specifying a certain amount of

[*] This research has been supported by DFG (project BR 1817/7-1) and FWF (projects I1102 and P25521).

[1] See (Jakobovits and Vermeir 1999; Arieli 2012; Grossi and Modgil 2015) for exemptions.

information. In a first step, we study the relation of these classes to each other. We therefore introduce the notion of being *more informative* capturing the intuition that a certain class can reproduce the information of an other. We present a hierarchy w.r.t. this ordering. The hierarchy contains 15 different verification classes only. This is due to the fact that many syntactically different classes collapse to the same amount of information.

We then formally define the essential property of a semantics σ being *verifiable* w.r.t. a certain verification class. We present a general theorem stating that any *rational* semantics is exactly verifiable w.r.t. one of the 15 different verification classes. Roughly speaking, a semantics is rational if attacks inbetween two self-loops can be omitted without affecting the set of extensions. An important aside hereby is that even the most informative class contains indeed less information than the entire framework by itself.

In this paper we consider a representative set of standard semantics. All of them satisfy rationality and thus, are exactly verifiable w.r.t. a certain class. Since the theorem does not provide an answer to which verification class perfectly matches a certain rational semantics we study this problem one by one for any considered semantics. As a result, only 6 different classes are essential to classify the considered standard semantics.

In the last part of the paper we study an application of the concept of verifiability. More precisely, we address the question of strong equivalence for semantics lying inbetween known semantics, so-called intermediate semantics. Strong equivalence is the natural counterpart to ordinary equivalence in monotonic theories (see (Oikarinen and Woltran 2011; Baumann 2016) for abstract argumentation and (Maher 1986; Lifschitz, Pearce, and Valverde 2001; Turner 2004; Truszczynski 2006) for other nonmonotonic theories). We provide characterization theorems relying on the notion of verifiability and thus, contributing to a more abstract understanding of the different features argumentation semantics offer. Besides these main results, we also give new characterizations for strong equivalence with respect to naive extensions and strong admissible sets.

Preliminaries

An *argumentation framework* (AF) $F = (A, R)$ is a directed graph whose nodes $A \subseteq \mathcal{U}$ (with $\mathcal{U}$ being an infinite set of arguments, so-called *universe*) are interpreted as *arguments* and whose edges $R \subseteq A \times A$ represent *conflicts* between them. We assume that all AFs possess finitely[2] many arguments only and denote the collection of all AFs by $\mathscr{A}$. If $(a, b) \in R$ we say that a *attacks* b. Alternatively, we write $a \rightarrowtail b$ as well as, for some $S \subseteq A$, $a \rightarrowtail S$ or $S \rightarrowtail b$ if there is some $c \in S$ attacked by a or attacking b, respectively. An argument $a \in A$ is *defended* by a set $S \subseteq A$ if for each $b \in A$ with $b \rightarrowtail a$, $S \rightarrowtail b$. We define the *range* of S (in F) as $S_F^+ = S \cup \{a \mid S \rightarrowtail a\}$. Similarly, we use S_F^- to denote the *anti-range* of S (in F) as $S \cup \{a \mid a \rightarrowtail S\}$. Furthermore, we say that a set S is *conflict-free* (in F) if there is no argument $a \in S$ s.t. $S \rightarrowtail a$. The set of all conflict-free sets of an AF F is denoted by $cf(F)$. For an AF $F = (B, S)$ we use $A(F)$ and $R(F)$ to refer to B and S, respectively. Furthermore, we use $L(F) = \{a \mid (a, a) \in R(F)\}$ for the set of all self-defeating arguments. Finally, we introduce the union of AFs F and G as $F \cup G = (A(F) \cup A(G), R(F) \cup R(G))$.

Semantics

A *semantics* σ assigns to each $F = (A, R)$ a set $\sigma(F) \subseteq 2^A$ where the elements are called *σ-extensions*. Numerous semantics are available. Each of them captures different intuitions about how to reason about conflicting knowledge. We consider $\sigma \in \{ad, na, stb, pr, co, gr, ss, stg, id, eg\}$ for admissible, naive, stable, preferred, complete, grounded, semi-stable, stage, ideal, and eager semantics (Dung 1995; Caminada, Carnielli, and Dunne 2012; Verheij 1996; Dung, Mancarella, and Toni 2007; Caminada 2007).

Definition 1. Given an AF $F = (A, R)$ and let $S \subseteq A$.

1. $S \in ad(F)$ iff $S \in cf(F)$ and each $a \in S$ is defended by S,
2. $S \in na(F)$ iff $S \in cf(F)$ and there is no $S' \in cf(F)$ s.t. $S \subsetneq S'$,
3. $S \in stb(F)$ iff $S \in cf(F)$ and $S_F^+ = A$,
4. $S \in pr(F)$ iff $S \in ad(F)$ and there is no $S' \in ad(F)$ s.t. $S \subsetneq S'$,
5. $S \in co(F)$ iff $S \in ad(F)$ and for any $a \in A$ defended by S, $a \in S$,
6. $S \in gr(F)$ iff $S \in co(F)$ and there is no $S' \in co(F)$ s.t. $S' \subsetneq S$,
7. $S \in ss(F)$ iff $S \in ad(F)$ and there is no $S' \in ad(F)$ s.t. $S_F^+ \subsetneq S'^+_F$,
8. $S \in stg(F)$ iff $S \in cf(F)$ and there is no $S' \in cf(F)$ s.t. $S_F^+ \subsetneq S'^+_F$,
9. $S \in id(F)$ iff $S \in ad(F)$, $S \subseteq \bigcap pr(F)$ and there is no $S' \in ad(F)$ satisfying $S' \subseteq \bigcap pr(F)$ s.t. $S \subsetneq S'$,
10. $S \in eg(F)$ iff $S \in ad(F)$, $S \subseteq \bigcap ss(F)$ and there is no $S' \in ad(F)$ satisfying $S' \subseteq \bigcap ss(F)$ s.t. $S \subsetneq S'$.

For two semantics σ, τ we use $\sigma \subseteq \tau$ to indicate that $\sigma(F) \subseteq \tau(F)$ for each AF $F \in \mathscr{A}$. If we have $\rho \subseteq \sigma$ and $\sigma \subseteq \tau$ for semantics ρ, σ, τ, we say that σ is *ρ-τ-intermediate*. Well-known relations between semantics are $stb \subseteq ss \subseteq pr \subseteq co \subseteq ad$, meaning, for instance, that ss is stb-pr-intermediate.

Definition 2. We call a semantics σ *rational* if self-loop-chains are irrelevant. That is, for every AF F it holds that $\sigma(F) = \sigma(F^l)$, where $F^l = (A(F), R(F) \setminus \{(a, b) \in R(F) \mid (a, a), (b, b) \in R(F), a \neq b\})$.

Indeed, all semantics introduced in Definition 1 are rational. A prominent semantics that is based on conflict-free sets, but is not rational is the *cf2*-semantics (Baroni, Giacomin, and Guida 2005), since here chains of self-loops can have an influence on the SCCs of an AF (see also (Gaggl and Woltran 2013)).

[2] Finiteness of AFs is a common assumption in argumentation papers. A systematic study of the infinite case has begun quite recently (cf. (Baumann and Spanring 2015) for an overview).

Equivalence and Kernels

The following definition captures the two main notions of equivalence available for non-monotonic formalisms, namely *ordinary* (or *standard*) *equivalence* and *strong* (or *expansion*) *equivalence*. A detailed overview of equivalence notion including their relations to each other can be found in (Baumann and Brewka 2013; 2015).

Definition 3. Given a semantics σ. Two AFs F and G are

- *standard equivalent* w.r.t. σ ($F \equiv^{\sigma} G$) iff $\sigma(F) = \sigma(G)$,
- *expansion equivalent* w.r.t. σ ($F \equiv^{\sigma}_{E} G$) iff for all AFs H: $F \cup H \equiv^{\sigma} G \cup H$

Expansion equivalence can be decided syntactically via so-called *kernels* (Oikarinen and Woltran 2011). A kernel is a function $k : \mathscr{A} \mapsto \mathscr{A}$ mapping each AF F to another AF $k(F)$ (which we may also denote as F^k). Consider the following definitions.

Definition 4. Given an AF $F = (A, R)$ and a semantics σ. We define σ-kernels $F^{k(\sigma)} = \left(A, R^{k(\sigma)}\right)$ whereby

$$\begin{aligned}
R^{k(stb)} &= R \setminus \{(a,b) \mid a \neq b, (a,a) \in R\},\\
R^{k(ad)} &= R \setminus \{(a,b) \mid a \neq b, (a,a) \in R,\\
&\qquad\qquad \{(b,a),(b,b)\} \cap R \neq \emptyset\},\\
R^{k(gr)} &= R \setminus \{(a,b) \mid a \neq b, (b,b) \in R,\\
&\qquad\qquad \{(a,a),(b,a)\} \cap R \neq \emptyset\},\\
R^{k(co)} &= R \setminus \{(a,b) \mid a \neq b, (a,a),(b,b) \in R\}.
\end{aligned}$$

We say that a relation $\equiv \subseteq \mathscr{A} \times \mathscr{A}$ is *characterizable through kernels* if there is a kernel k, s.t. $F \equiv G$ iff $F^k = G^k$. Moreover, we say that a semantics σ is *compatible with a kernel* k if $F \equiv^{\sigma}_{E} G$ iff $F^k = G^k$. All semantics (except naive semantics) considered in this paper are compatible with one of the four kernels introduced above. In the next section, we will complete these results taking naive semantics and strong admissible sets into account.

Theorem 1. *(Oikarinen and Woltran 2011; Baumann and Woltran 2014) For any AFs F and G,*

1. $F \equiv^{\sigma}_{E} G \Leftrightarrow F^{k(\sigma)} = G^{k(\sigma)}$ *with* $\sigma \in \{stb, ad, co, gr\}$,
2. $F \equiv^{\tau}_{E} G \Leftrightarrow F^{k(ad)} = G^{k(ad)}$ *with* $\tau \in \{pr, id, ss, eg\}$,
3. $F \equiv^{stg}_{E} G \Leftrightarrow F^{k(stb)} = G^{k(stb)}$.

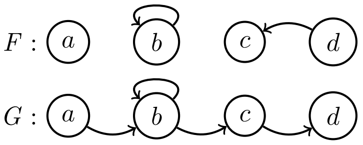
Complementing Previous Results

In order to provide an exhaustive analysis of intermediate semantics (confer penultimate section) we provide missing kernels for naive semantics as well as strongly admissible sets. We start with the so-called *naive kernel* characterizing expansion equivalence w.r.t. naive semantics. As an aside, the following kernel is the first one which adds attacks to the former attack relation.

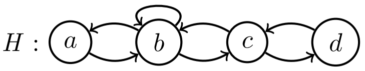
Definition 5. Given an AF $F = (A, R)$. We define the *naive kernel* $F^{k(na)} = \left(A, R^{k(na)}\right)$ whereby $R^{k(na)} = R \cup \{(a,b) \mid a \neq b, \{(a,a),(b,a),(b,b)\} \cap R \neq \emptyset\}$.

The following example illustrates the definition above.

Example 1. Consider the AFs F and G. Note that $na(F) = na\,(G) = \{\{a,c\},\{a,d\}\}$. Consequently, $F \equiv^{na} G$.



In accordance with Definition 5 we observe that both AFs possess the same naive kernel $H = F^{k(na)} = G^{k(na)}$.



The following theorem proves that possessing the same kernels is necessary as well as sufficient for being strongly equivalent, i.e. $F \equiv^{na}_{E} G$.

Theorem 2. *For all AFs F, G,*

$$F \equiv^{na}_{E} G \Leftrightarrow F^{k(na)} = G^{k(na)}.$$

Proof. In (Baumann and Woltran 2014) it was already shown that $F \equiv^{na}_{E} G$ iff jointly $A(F) = A(G)$ and $na(F) = na(G)$. Consequently, it suffices to prove that $F^{k(na)} = G^{k(na)}$ implies $A(F) = A(G)$ as well as $na(F) = na(G)$ and vice versa.

($\Leftarrow$) Given $F^{k(na)} = G^{k(na)}$. By Definition 5 we immediately have $A(F) = A(G)$. Assume now that $na(F) \neq na(G)$ and without loss of generality let $S \in na(F) \setminus na(G)$. Obviously, for any AF H, $cf(H) = cf\left(H^{k(na)}\right)$. Hence, there is an S', s.t. $S \subsetneq S' \in cf(G) \setminus cf(F)$. Thus, there are $a, b \in S' \setminus S$, s.t. $(a,b) \in R(F) \setminus R(G)$. Furthermore, $(a,a),(b,b) \notin R(G)$ and since for any AF H, $L(H) = L\left(H^{k(na)}\right)$ we obtain $(a,a),(b,b) \notin R(F)$. Consequently, we have to consider $a \neq b$. Since $(a,b) \in R(F) \setminus R(G)$, we obtain $(a,b),(b,a) \in R\left(F^{k(na)}\right)$. Since $F^{k(na)} = G^{k(na)}$ is assumed we derive $(a,b),(b,a) \in R\left(G^{k(na)}\right)$. By Definition 5 we must have $(b,a) \in R(G)$ contradicting the conflict-freeness of S' in G.

($\Rightarrow$) We show the contrapositive, i.e. $F^{k(na)} \neq G^{k(na)}$ implies $A(F) \neq A(G)$ or $na(F) \neq na(G)$. Observe that for any AF H, $A(H) = A\left(H^{k(na)}\right)$. Consequently, if $A\left(F^{k(na)}\right) \neq A\left(G^{k(na)}\right)$, then $A(F) \neq A(G)$. Assume now $R\left(F^{k(na)}\right) \neq R\left(G^{k(na)}\right)$. Without loss of generality let $(a,b) \in R\left(F^{k(na)}\right) \setminus R\left(G^{k(na)}\right)$. Since for any AF H, $L(H) = L\left(H^{k(na)}\right)$ we obtain $a \neq b$. Furthermore, $(a,b) \in R\left(F^{k(na)}\right)$ implies $\{(a,a),(a,b),(b,a),(b,b)\} \cap R(F) \neq \emptyset$ and consequently, for any $S \in na(F)$, $\{a,b\} \not\subseteq S$. Since $(a,b) \notin R\left(G^{k(na)}\right)$ we deduce $\{(a,a),(a,b),(b,a),(b,b)\} \cap R(F) = \emptyset$. Hence, $\{a,b\} \in cf(G)$ and thus, there exists a set $S \in na(G)$, s.t. $\{a,b\} \subseteq S$ (compare (Baumann and Spanring 2015, Lemma 3)) witnessing $na(F) \neq na(G)$. □

We turn now to *strongly admissible sets* (for short, *sad*) (Baroni and Giacomin 2007b). We will show that, beside grounded (Oikarinen and Woltran 2011) and resolution based grounded semantics (Baroni, Dunne, and Giacomin 2011; Dvořák et al. 2014), strongly admissible sets are characterizable through the grounded kernel. Consider the following self-referential definition taken from (Caminada 2014).

Definition 6. Given an AF $F = (A, R)$. A set $S \subseteq A$ is *strongly admissible*, i.e. $S \in \mathit{sad}(F)$ iff any $a \in S$ is defended by a strongly admissible set $S' \subseteq S \setminus \{a\}$.

The following properties are needed to prove the characterization theorem. The first two of them are already shown in (Baroni and Giacomin 2007a). The third statement is an immediate consequence of the former.

Proposition 1. *Given two AFs F and G, then*

1. *$gr(F) \subseteq sad(F) \subseteq ad(F)$,*
2. *if $S \in gr(F)$ we have: $S' \subseteq S$ for all $S' \in sad(F)$, and*
3. *$sad(F) = sad(G)$ implies $gr(F) = gr(G)$.*

The following definition provides us with an alternative criterion for being a strong admissible set. In contrast to the former it allows one to construct strong admissible sets step by step. Thus, a construction method is given.

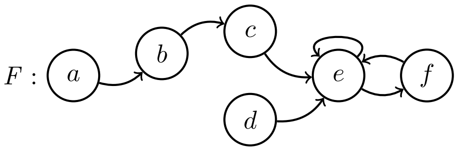
Definition 7. Given an AF $F = (A, R)$. A set $S \subseteq A$ is *strongly admissible*, i.e. $S \in \mathit{sad}(F)$ iff there are finitely many and pairwise disjoint sets $A_1, ..., A_n$, s.t. $S = \bigcup_{1 \le i \le n} A_i$ and $A_1 \subseteq \Gamma_F(\emptyset)$[3] and furthermore, $\bigcup_{1 \le i \le j} A_i$ defends A_{j+1} for $1 \le j \le n-1$.

Proposition 2. *Definitions 6 and 7 are equivalent.*

Proof. For the proof we use $S \in \mathit{sad}_k(F)$ as a shorthand for $S \in \mathit{sad}(F)$ in the sense of Definition k. ($\Leftarrow$) Given $S \in \mathit{sad}_7(F)$. Hence, there is a finite partition, s.t. $S = \bigcup_{1 \le i \le n} A_i$, $A_1 \subseteq \Gamma_F(\emptyset)$ and $\bigcup_{1 \le i \le j} A_i$ defends A_{j+1} for $1 \le j \le n-1$. Observe that $\bigcup_{1 \le i \le j} A_i \in \mathit{sad}_7(F)$ for any $j \le n$. Let $a \in S$. Consequently, there is an index i^*, s.t. $a \in A_{i^*}$. Furthermore, since $\bigcup_{1 \le i \le i^*-1} A_i$ defends A_{i^*} by definition, we deduce that $\bigcup_{1 \le i \le i^*-1} A_i \subseteq S \setminus \{a\}$ defends a. We have to show now that (the smaller set w.r.t. $\subseteq$) $\bigcup_{1 \le i \le i^*-1} A_i \in \mathit{sad}_6(F)$. Note that $\bigcup_{1 \le i \le i^*-1} A_i \in \mathit{sad}_7(F)$. Since we are dealing with finite AFs we may iterate our construction. Hence, no matter which elements are chosen we end up with a $\subseteq$-chain, s.t. $\emptyset \subseteq \bigcup_{1 \le i \le i_e} A_i \subseteq S_e \setminus a_e$ and $\emptyset$ defends a_e for some index i_e, set S_e and element a_e. This means, the question whether $S \in \mathit{sad}_6(F)$ can be decided positively by proving $\emptyset \in \mathit{sad}_6(F)$. Since the empty set does not contain any elements we find $\emptyset \in \mathit{sad}_6(F)$ concluding $\mathit{sad}_7 \subseteq \mathit{sad}_6$.
($\Rightarrow$) Given $S \in \mathit{sad}_6(F)$, consider the following sets S_i: $S_1 = (\Gamma(\emptyset) \setminus \emptyset) \cap S$, $S_2 = (\Gamma(S_1) \setminus S_1) \cap S$, $S_3 = \left(\Gamma(\bigcup_{i=1}^{2} S_i) \setminus \bigcup_{i=1}^{2} S_i\right) \cap S$, $\ldots$, $S_n = \left(\Gamma(\bigcup_{i=1}^{n-1} S_i) \setminus \bigcup_{i=1}^{n-1} S_i\right) \cap S$. Since we are dealing with finite AFs there has to be a natural $n \in \mathbb{N}$, s.t. $S_n = S_{n+1} = S_{n+2} = \ldots$. Consider now the union of these sets, i.e. $\bigcup_{i=1}^{n} S_i$. We show now that $\bigcup_{i=1}^{n} S_i \in \mathit{sad}_7(F)$ and $\bigcup_{i=1}^{n} S_i = S$. By construction we have $S_1 \subseteq \Gamma(\emptyset)$. Moreover, $\bigcup_{1 \le i \le j} S_i$ defends S_{j+1} for $1 \le j \le n-1$. This can be seen as follows. By definition $S_{j+1} = \left(\Gamma(\bigcup_{i=1}^{j} S_i) \setminus \bigcup_{i=1}^{j} S_i\right) \cap S$. This means, $S_{j+1} \subseteq \Gamma(\bigcup_{i=1}^{j} S_i)$. Since $\Gamma(\bigcup_{i=1}^{j} S_i)$ contains all elements defended by $\bigcup_{i=1}^{j} S_i$ we obtain $\bigcup_{i=1}^{n} S_i \in \mathit{sad}_7(F)$. Obviously, $\bigcup_{i=1}^{j} S_i \subseteq S$. In order to derive a contradiction we suppose $S \not\subseteq \bigcup_{i=1}^{n} S_i$. This means there is a nonempty set S^*, s.t. $S = S^* \cup \bigcup_{i=1}^{n} S_i$. Let $S^* = \{s_1, \ldots, s_k\}$. Observe that no element s_i is defended by $\bigcup_{i=1}^{n} S_i$ (*). Since $S \in \mathit{sad}_6(F)$ we obtain a set $S_1^* \subseteq S \setminus \{s_1\}$, s.t. $S_1^* \in \mathit{sad}_6(F)$ and S_1^* defends s_1. We now iterate this procedure ending up with a set $S_k^* \subseteq S_{k-1}^* \setminus \{s_k\} \subseteq \bigcup_{i=1}^{n} S_i$, s.t. $S_k^* \in \mathit{sad}_6(F)$ and S_k^* defends s_k contradicting (*) and concluding the proof. □

The following example shows how to use the new construction method.

Example 2. Consider the following AF F.



We have $\Gamma_F(\emptyset) = \{a, d\}$. Hence, for all $S \subseteq \{a, d\}$, $S \in \mathit{sad}(F)$. Furthermore, $\Gamma_F(\{a\}) = \{a, c\}$, $\Gamma_F(\{d\}) = \{d, f\}$ and $\Gamma_F(\{a, d\}) = \{a, d, c, f\}$. This means, additionally $\{a, c\}, \{d, f\}, \{a, d, c\}, \{a, d, f\}, \{a, d, c, f\} \in \mathit{sad}(F)$. Finally, $\Gamma_F(\{a, c\}) = \{a, c, f\}$ justifying the last missing set $\{a, c, f\} \in \mathit{sad}(F)$.

The following corollary is an immediate consequence of Definition 7. It is essential to prove the characterization theorem for strongly admissible sets.

Corollary 1. *Given an AF F and two sets $B, B' \subseteq A(F)$. If B defends B', then $B \cup B'$ is strong admissible if B is.*

The following lemma shows that the grounded kernel is insensitive w.r.t. strong admissible sets.

Lemma 1. *For any AF F, $sad(F) = sad\left(F^{k(gr)}\right)$.*

Proof. The grounded kernel is node- and loop-preserving, i.e. $A(F) = A\left(F^{k(gr)}\right)$ and $L(F) = L\left(F^{k(gr)}\right)$. Furthermore, $\mathit{cf}(F) = \mathit{cf}\left(F^{k(gr)}\right)$ and $\Gamma_F(\emptyset) = \Gamma_{F^{k(gr)}}(\emptyset)$ as shown in (Oikarinen and Woltran 2011, Lemma 6).
($\subseteq$) Given $S \in \mathit{sad}(F)$. The proof is by induction on n indicating the number of sets forming a suitable (according to Definition 7) partition of S. Let $n = 1$. In consideration of the grounded kernel we observe $\Gamma_F(\emptyset) = \Gamma_{F^{k(gr)}}(\emptyset)$, i.e. the set of unattacked arguments does not change. Since $S \subseteq \Gamma_F(\emptyset)$ is assumed we are done. Assume now that the assertion is proven for any k-partition. Let S be a $(k+1)$-partition, i.e. $S = \bigcup_{i=1}^{k+1} A_i$. According to induction hypothesis as well as Corollary 1 it suffices to prove $\bigcup_{i=1}^{k} A_i$ defends A_{k+1} in $F^{k(gr)}$. Assume not, i.e. there are arguments $b \in A(F) \setminus S$, $c \in A_{k+1}$ s.t. $(b, c) \in R\left(F^{k(gr)}\right) \subseteq R(F)$ and for all $a \in \bigcup_{i=1}^{k} A_i$, $(a, b) \notin R\left(F^{k(gr)}\right)$ (*). Since $\bigcup_{i=1}^{k} A_i$ defends A_{k+1} in F we deduce the existence of an argument $a \in \bigcup_{i=1}^{k} A_i$ s.t. $(a, b) \in R(F)$. Thus, (a, b) is redundant w.r.t. the grounded kernel. According to Definition 4 and due to

[3] Hereby, Γ is the so-called *characteristic function* (Dung 1995) with $\Gamma_F(S) = \{a \in A \mid a \text{ is defended by } S \text{ in } F\}$. The term $\Gamma_F(\emptyset)$ can be equivalently replaced by $\{a \in A \mid a \text{ is unattacked}\}$.

the conflict-freeness of $\bigcup_{i=1}^{k} A_i$ we have $(a,a) \notin R(F)$ and $(b,a),(b,b) \in R(F)$. Consequently, $(b,a) \in F^{k(gr)}$. Since $\bigcup_{i=1}^{k} A_i$ is a strong admissible k-partition in F we obtain by induction hypothesis that $\bigcup_{i=1}^{k} A_i$ is strong admissible in $F^{k(gr)}$ and therefore, admissible in $F^{k(gr)}$ (Proposition 1). Hence there has to be an argument $a \in \bigcup_{i=1}^{k} A_i$, s.t. $(a,b) \in R\left(F^{k(gr)}\right)$, contradicting (*).
($\supseteq$) Assume $S \in \mathit{sad}\left(F^{k(gr)}\right)$. We show $S \in \mathit{sad}(F)$ by induction on n indicating that S is a n-partition in $F^{k(gr)}$. Due to $\Gamma_F(\emptyset) = \Gamma_{F^{k(gr)}}(\emptyset)$ the base case is immediately clear. For the induction step let S be a $(k+1)$-partition, i.e. $S = \bigcup_{i=1}^{k+1} A_i$. By induction hypothesis we may assume that $\bigcup_{i=1}^{k} A_i$ is strongly admissible in F. Using Corollary 1 it suffices to prove $\bigcup_{i=1}^{k} A_i$ defends A_{k+1} in F. Assume not, i.e. there are arguments $b \in A(F) \setminus S$, $c \in A_{k+1}$ s.t. $(b,c) \in R(F)$ and for all $a \in \bigcup_{i=1}^{k} A_i$, $(a,b) \notin R(F)$. We even have $(a,b) \notin R\left(F^{k(gr)}\right)$ since $R\left(F^{k(gr)}\right) \subseteq R(F)$. Consequently, (b,c) has to be deleted in $F^{k(gr)}$. Definition 4 requires $(c,c) \in R\left(F^{k(gr)}\right)$ contradicting the conflict-freeness of S in $F^{k(gr)}$. □

Theorem 3. *For any two AFs F and G we have,*

$$F \equiv_E^{sad} G \Leftrightarrow F^{k(gr)} = G^{k(gr)}$$

Proof. ($\Rightarrow$) We show the contrapositive, i.e. $F^{k(gr)} \neq G^{k(gr)} \Rightarrow F \not\equiv_E^{sad} G$. Assuming $F^{k(gr)} \neq G^{k(gr)}$ implies $F \not\equiv_E^{gr} G$ (Theorem 1). This means, there is an AF H, s.t. $\mathit{gr}(F \cup H) \neq \mathit{gr}(G \cup H)$. Due to statement 3 of Proposition 1, we deduce $\mathit{sad}(F \cup H) \neq \mathit{sad}(G \cup H)$ proving $F \not\equiv_E^{sad} G$.
($\Leftarrow$) Given $F^{k(gr)} = G^{k(gr)}$. Since expansion equivalence is a congruence w.r.t. $\cup$ we obtain $(F \cup H)^{k(gr)} = (G \cup H)^{k(gr)}$ for any AF H. Consequently, $\mathit{sad}\left((F \cup H)^{k(gr)}\right) = \mathit{sad}\left((G \cup H)^{k(gr)}\right)$. Due to Lemma 1 we deduce $\mathit{sad}(F \cup H) = \mathit{sad}(G \cup H)$, concluding the proof. □

Verifiability

In this section we study the question whether we really need the entire AF F to compute the extensions of a given semantics. Let us consider naive semantics. Obviously, in order to determine naive extensions it suffices to know all conflict-free sets. Conversely, knowing $\mathit{cf}(F)$ only does not allow to reconstruct F unambiguously. This means, knowledge about $\mathit{cf}(F)$ is indeed less information than the entire AF by itself. In fact, most of the existing semantics do not need information of the entire framework. We will categorize the amount of information by taking the conflict-free sets as a basis and distinguish between different amounts of knowledge about the neighborhood, that is range and anti-range, of these sets.

Definition 8. We call a function $\mathfrak{r}^x : 2^{\mathcal{U}} \times 2^{\mathcal{U}} \to \left(2^{\mathcal{U}}\right)^n$ $(n > 0)$ which is expressible via basic set operations only *neighborhood function*. A neighborhood function $\mathfrak{r}^x$ induces the *verification class* mapping each AF F to

$$\widetilde{F}^x = \{(S, \mathfrak{r}^x(S_F^+, S_F^-)) \mid S \in \mathit{cf}(F)\}.$$

We coined the term neighborhood function because the induced verification classes apply these functions to the neighborhoods, i.e. range and anti-range of conflict-free sets. The notion of *expressible via basic set operations* simply means that (in case of $n = 1$) the expression $\mathfrak{r}^x(A,B)$ is in the language generated by the following BNF:

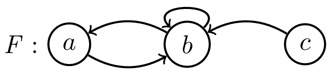
$$X ::= A \mid B \mid (X \cup X) \mid (X \cap X) \mid (X \setminus X).$$

Consequently, in case of $n = 1$, we may distinguish eight set theoretically different neighborhood functions, namely

$$\begin{aligned}
\mathfrak{r}^{\epsilon}(S,S') &= \emptyset \\
\mathfrak{r}^{+}(S,S') &= S \\
\mathfrak{r}^{-}(S,S') &= S' \\
\mathfrak{r}^{\mp}(S,S') &= S' \setminus S \\
\mathfrak{r}^{\pm}(S,S') &= S \setminus S' \\
\mathfrak{r}^{\cap}(S,S') &= S \cap S' \\
\mathfrak{r}^{\cup}(S,S') &= S \cup S' \\
\mathfrak{r}^{\triangle}(S,S') &= (S \cup S') \setminus (S \cap S')
\end{aligned}$$

A verification class encapsulates a certain amount of information about an AF, as the following example illustrates.

Example 3. Consider the following AF F:



Now take, for instance, the verification class induced by $\mathfrak{r}^+$, that is $\widetilde{F}^+ = \{(S, \mathfrak{r}^+(S_F^+, S_F^-)) \mid S \in \mathit{cf}(F)\} = \{(S, S_F^+) \mid S \in \mathit{cf}(F)\}$, storing information about conflict-free sets together with their associated ranges w.r.t. F. It contains the following tuples: $(\emptyset, \emptyset)$, $(\{a\},\{b\})$, $(\{c\},\{b\})$, and $(\{a,c\},\{b\})$. The verification class induced by $\mathfrak{r}^{\pm}$ contains the same tuples but $(\{a\},\emptyset)$ instead of $(\{a\},\{b\})$.

Intuitively, it should be clear that the set $\widetilde{F}^+$ suffices to compute stage extensions (i.e., range-maximal conflict-free sets) of F. This intuitive understanding of *verifiability* will be formally specified in Definition 10. Note that a neighborhood function $\mathfrak{r}^x$ may return n-tuples. Consequently, in consideration of the eight listed basic function we obtain (modulo reordering, duplicates, empty set) $2^7 + 1$ syntactically different neighborhood functions and therefore the same number of verification classes. As usual, we will denote the n-ary combination of basic functions $(\mathfrak{r}^{x_1}(S,S'), \ldots, \mathfrak{r}^{x_n}(S,S'))$ as $\mathfrak{r}^x(S,S')$ with $x = x_1 \ldots x_n$.

With the following definition we can put neighborhood functions into relation w.r.t. their information. This will help us to show that actually many of the induced classes collapse to the same amount of information.

Definition 9. Given neighborhood functions $\mathfrak{r}^x$ and $\mathfrak{r}^y$ returning n-tuples and m-tuples, respectively, we say that $\mathfrak{r}^x$ is *more informative* than $\mathfrak{r}^y$, for short $\mathfrak{r}^x \succeq \mathfrak{r}^y$, iff there is a function $\delta : \left(2^{\mathcal{U}}\right)^n \to \left(2^{\mathcal{U}}\right)^m$ such that for any two sets of arguments $S, S' \subseteq \mathcal{U}$, we have $\delta\left(\mathfrak{r}^x(S,S')\right) = \mathfrak{r}^y\left(S,S'\right)$.

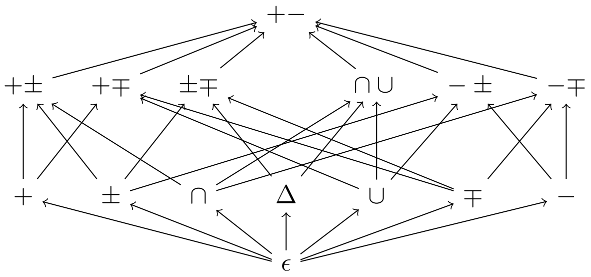


Figure 1: Representatives of neighborhood functions and their relation w.r.t. information; a node x stands for the neighborhood function $\mathfrak{r}^x$; an arrow from x to y means $\mathfrak{r}^x \prec \mathfrak{r}^y$.

We will denote the strict part of $\succeq$ by $\succ$, i.e. $\mathfrak{r}^x \succ \mathfrak{r}^y$ iff $\mathfrak{r}^x \succeq \mathfrak{r}^y$ and $\mathfrak{r}^y \not\succeq \mathfrak{r}^x$. Moreover $\mathfrak{r}^x \approx \mathfrak{r}^y$ in case $\mathfrak{r}^x \succeq \mathfrak{r}^y$ and $\mathfrak{r}^y \succeq \mathfrak{r}^x$, we say that $\mathfrak{r}^x$ *represents* $\mathfrak{r}^y$ and vice versa.

Lemma 2. *All neighborhood functions are represented by the ones depicted in Figure 1 and the $\prec$-relation represented by arcs in Figure 1 holds.*

Proof. We begin by showing that all neighborhood functions are represented in Figure 1. Clearly, each neighborhood function $\mathfrak{r}^x$ represents itself, i.e. $\mathfrak{r}^x \approx \mathfrak{r}^x$. All neighborhood functions for $n = 1$ are are depicted in Figure 1. We turn to $n = 2$. Consider the neighborhood functions $\mathfrak{r}^{+\pm}$, $\mathfrak{r}^{+\cap}$, and $\mathfrak{r}^{\pm\cap}$, defined as $\mathfrak{r}^{+\pm}(S, S') = (S, S \setminus S')$, $\mathfrak{r}^{+\cap}(S, S') = (S, S \cap S')$, and $\mathfrak{r}^{\pm\cap}(S, S') = (S \setminus S', S \cap S')$ for $S, S' \subseteq \mathcal{U}$. Observe that $S = (S \setminus S') \cup (S \cap S')$. Hence, we can easily define functions in the spirit of Definition 9 mapping the images of the function to one another:

- $\delta_1(\mathfrak{r}^{+\pm}(S, S')) = \delta_1(S, S \setminus S') =_{def} (S, S \setminus (S \setminus S')) = (S, S \cap S') = \mathfrak{r}^{+\cap}(S, S')$;
- $\delta_2(\mathfrak{r}^{+\cap}(S, S')) = \delta_2(S, S \cap S') =_{def} (S \setminus (S \cap S'), S \cap S') = (S \setminus S', S \cap S') = \mathfrak{r}^{\pm\cap}(S, S')$;
- $\delta_3(\mathfrak{r}^{\pm\cap}(S, S')) = \delta_3(S \setminus S', S \cap S') =_{def} ((S \setminus S') \cup (S \cap S'), S \setminus S') = (S, S \setminus S') = \mathfrak{r}^{+\pm}(S, S')$.

Therefore, $\mathfrak{r}^{+\pm} \approx \mathfrak{r}^{+\cap} \approx \mathfrak{r}^{\pm\cap}$. In particular, they are all represented by $\mathfrak{r}^{\pm}$. We can apply the same reasoning to other combinations of neighborhood functions and get the following equivalences w.r.t. information content: $\mathfrak{r}^{+\mp} \approx \mathfrak{r}^{+\cup} \approx \mathfrak{r}^{\mp\cup}$; $\mathfrak{r}^{\pm\mp} \approx \mathfrak{r}^{\pm\Delta} \approx \mathfrak{r}^{\mp\Delta}$; $\mathfrak{r}^{\cap\cup} \approx \mathfrak{r}^{\cap\Delta} \approx \mathfrak{r}^{\cup\Delta}$; $\mathfrak{r}^{-\pm} \approx \mathfrak{r}^{-\cup} \approx \mathfrak{r}^{\pm\cup}$; and $\mathfrak{r}^{-\mp} \approx \mathfrak{r}^{-\cap} \approx \mathfrak{r}^{\mp\cap}$, with the functions stated first acting as representatives in Figure 1.

For the remaining functions returning 2-tuples we get $\mathfrak{r}^{+-} \approx \mathfrak{r}^{+\Delta} \approx \mathfrak{r}^{-\Delta}$ by

- $\delta_4(\mathfrak{r}^{+-}(S, S')) = \delta_4(S, S') =_{def} (S, (S \cup S') \setminus (S \cap S')) = \mathfrak{r}^{+\Delta}(S, S')$;
- $\delta_5(\mathfrak{r}^{+\Delta}(S, S')) = \delta_5(S, (S \cup S') \setminus (S \cap S')) =_{def} ((S \setminus ((S \cup S') \setminus (S \cap S'))) \cup ((S \cup S') \setminus (S \cap S')) \setminus S, (S \cup S') \setminus (S \cap S')) = (S', (S \cup S') \setminus (S \cap S')) = \mathfrak{r}^{-\cap}(S, S')$;
- $\delta_6(\mathfrak{r}^{-\Delta}(S, S')) = \delta_6(S', (S \cup S') \setminus (S \cap S')) =_{def} ((S' \setminus ((S \cup S') \setminus (S \cap S'))) \cup ((S \cup S') \setminus (S \cap S')) \setminus S', S') = (S, S') = \mathfrak{r}^{+-}(S, S')$.

Finally, every neighborhood function $\mathfrak{r}^{x_1 \cdots x_n}$ with $n \geq 3$ is represented by $\mathfrak{r}^{+-}$ since we can compute all possible sets from S and S'.

Now consider two functions $\mathfrak{r}^x$ and $\mathfrak{r}^y$ such that there is an arrow from x to y in Figure 1. It is easy to see that $\mathfrak{r}^y \succeq \mathfrak{r}^x$ since, for sets of arguments S and S', $\mathfrak{r}^x(S, S')$ is either contained in $\mathfrak{r}^y(S, S')$ or obtainable from $\mathfrak{r}^y(S, S')$ by basic set operations. The fact that $\mathfrak{r}^x \not\succeq \mathfrak{r}^y$, entailing $\mathfrak{r}^y \succ \mathfrak{r}^x$, follows from the impossibility of finding a function δ such that $\delta(\mathfrak{r}^x(S, S')) = \mathfrak{r}^y(S, S')$. □

If the information provided by a neighborhood function is sufficient to compute the extensions, we say the semantics is verifiable by the class induced by the neighborhood function.

Definition 10. A semantics σ is *verifiable* by the verification class induced by the neighborhood function $\mathfrak{r}^x$ returning n-tuples (or simply, x*-verifiable*) iff there is a function (also called *criterion*) $\gamma_\sigma : \left(2^{\mathcal{U}}\right)^n \times 2^{\mathcal{U}} \to 2^{2^{\mathcal{U}}}$ s.t. for every AF $F \in \mathscr{A}$ we have:

$$\gamma_\sigma\left(\widetilde{F}^x, A(F)\right) = \sigma(F).$$

Moreover, σ is *exactly x-verifiable* iff σ is x-verifiable and there is no verification class induced by $\mathfrak{r}^y$ with $\mathfrak{r}^y \prec \mathfrak{r}^x$ such that σ is y-verifiable.

Observe that if a semantics σ is x-verifiable then for any two AFs F and G with $\widetilde{F}^x = \widetilde{G}^x$ and $A(F) = A(G)$ it must hold that $\sigma(F) = \sigma(G)$.

We proceed with a list of criteria showing that any semantics mentioned in Definition 1 is verifiable by a verification class induced by a certain neighborhood function. In the following, we abbreviate the tuple $(\widetilde{F}^x, A(F))$ by $\widetilde{F}^x_A$.

$$\begin{aligned}
\gamma_{na}(\widetilde{F}^\epsilon_A) &= \{S \mid S \in \widetilde{F}, S \textit{ is } \subseteq\textit{-maximal in } \widetilde{F}\};\\
\gamma_{stg}(\widetilde{F}^+_A) &= \{S \mid (S, S^+) \in \widetilde{F}^+, S^+ \textit{ is } \subseteq\textit{-maximal in}\\
&\qquad \{C^+ \mid (C, C^+) \in \widetilde{F}^+\}\};\\
\gamma_{stb}(\widetilde{F}^+_A) &= \{S \mid (S, S^+) \in \widetilde{F}^+, S^+ = A\};\\
\gamma_{ad}(\widetilde{F}^\mp_A) &= \{S \mid (S, S^\mp) \in \widetilde{F}^\mp, S^\mp = \emptyset\};\\
\gamma_{pr}(\widetilde{F}^\mp_A) &= \{S \mid S \in \gamma_{ad}(\widetilde{F}^\mp_A), S \textit{ is } \subseteq\textit{-maximal in } \gamma_{ad}(\widetilde{F}^\mp_A)\};\\
\gamma_{ss}(\widetilde{F}^{+\mp}_A) &= \{S \mid S \in \gamma_{ad}(\widetilde{F}^\mp_A), S^+ \textit{ is } \subseteq\textit{-maximal in}\\
&\qquad \{C^+ \mid (C, C^+, C^\mp) \in \widetilde{F}^{+\mp}, C \in \gamma_{ad}(\widetilde{F}^\mp_A)\}\};\\
\gamma_{id}(\widetilde{F}^\mp_A) &= \{S \mid S \textit{ is } \subseteq\textit{-maximal in}\\
&\qquad \{C \mid C \in \gamma_{ad}(\widetilde{F}^\mp_A), C \subseteq \bigcap \gamma_{pr}(\widetilde{F}^\mp_A)\}\};\\
\gamma_{eg}(\widetilde{F}^{+\mp}_A) &= \{S \mid S \textit{ is } \subseteq\textit{-maximal in}\\
&\qquad \{C \mid C \in \gamma_{ad}(\widetilde{F}^\mp_A), C \subseteq \bigcap \gamma_{ss}(\widetilde{F}^{+\mp}_A)\}\};\\
\gamma_{sad}(\widetilde{F}^{-\pm}_A) &= \{S \mid (S, S^-, S^\pm) \in \widetilde{F}^{-\pm},\\
&\qquad \exists (S_0, S_0^-, S_0^\pm), \ldots, (S_n, S_n^-, S_n^\pm) \in \widetilde{F}^{-\pm} :\\
&\qquad (\emptyset = S_0 \subset \cdots \subset S_n = S \wedge\\
&\qquad \forall i \in \{1, \ldots, n\} : S_i^- \subseteq S_{i-1}^\pm)\};
\end{aligned}$$

$$\begin{aligned}\gamma_{gr}(\widetilde{F}_A^{-\pm}) = \{S \mid S \in \gamma_{sad}(\widetilde{F}_A^{-\pm}),\\ \forall(\bar{S}, \bar{S}^-, \bar{S}^{\pm}) \in \widetilde{F}^{-\pm} : \bar{S} \supset S \Rightarrow (\bar{S}^- \setminus S^{\pm}) \neq \emptyset)\};\\ \gamma_{co}(\widetilde{F}_A^{+-}) = \{S \mid (S, S^+, S^-) \in \widetilde{F}^{+-}, (S^- \setminus S^+) = \emptyset,\\ \forall(\bar{S}, \bar{S}^+, \bar{S}^-) \in \widetilde{F}^{+-} : \bar{S} \supset S \Rightarrow (\bar{S}^- \setminus S^+) \neq \emptyset)\}.\end{aligned}$$

Instead of a formal proof we give the following explanations. First of all it is easy to see that the naive semantics is verifiable by the verification class induced by $\mathfrak{r}^{\epsilon}$ since the naive extensions can be determined by the conflict-free sets. Stable and stage semantics, on the other hand, utilize the range of each conflict-free set in addition. Hence they are verifiable by the verification class induced by $\mathfrak{r}^{+}$. Now consider admissible sets. Recall that a conflict-free S set is admissible if and only if it attacks all attackers. This is captured exactly by the condition $S^{\mp} = \emptyset$, hence admissible sets are verifiable by the verification class induced by $\mathfrak{r}^{\mp}$. The same holds for preferred semantics, since we just have to determine the maximal conflict-free sets with $S^{\mp} = \emptyset$. Semi-stable semantics, however, needs the range of each conflict-free set in addition, see γ_{ss}, which makes it verifiable by the verification class induced by $\mathfrak{r}^{+\mp}$. Finally consider the criterion γ_{co}. The first two conditions for a set of arguments S stand for conflict-freeness and admissibility, respectively. Now assume the third condition does not hold, i.e., there exists a tuple $(\bar{S}, \bar{S}^+, \bar{S}^-) \in \widetilde{F}^{+-}$ with $\bar{S} \supset S$ and $\bar{S}^- \setminus S^+ = \emptyset$. This means that every argument attacking $\bar{S}$ is attacked by S, i.e., $\bar{S}$ is defended by S. Hence S is not a complete extension, showing that $\gamma_{co}(\widetilde{F}_A^{+-}) = co(F)$ for each $F \in \mathscr{A}$. One can verify that all criteria from the list are adequate in the sense that they describe the extensions of the corresponding semantics.

We show now that the formal concepts of verifiability and being more informative behave correctly in the sense that the use of more informative neighborhood functions do not lead to a loss of verification capacity.

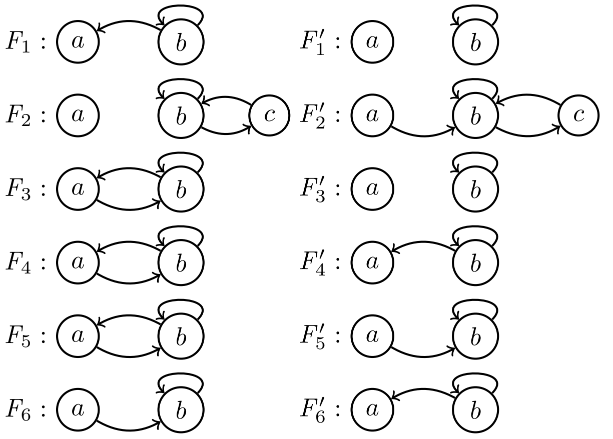
Proposition 3. *If a semantics σ is x-verifiable, then σ is verifiable by all verification classes induced by some $\mathfrak{r}^y$ with $\mathfrak{r}^y \succeq \mathfrak{r}^x$.*

Proof. As σ is verifiable by the verification class induced by $\mathfrak{r}^x$ it holds that there is some γ_σ such that for all $F \in \mathscr{A}$, $\gamma_\sigma(\widetilde{F}^x, A(F)) = \sigma(F)$. Now let $\mathfrak{r}^y \succeq \mathfrak{r}^x$, meaning that there is some δ such that $\delta(\mathfrak{r}^y(S, S')) = \mathfrak{r}^x$. We define $\gamma'_\sigma(\widetilde{F}^y, A(F)) = \gamma_\sigma(\{(S, \delta(\mathcal{S})) \mid (S, \mathcal{S}) \in \widetilde{F}^y\}, A(F))$ and observe that $\{(S, \delta(\mathcal{S})) \mid (S, \mathcal{S}) \in \widetilde{F}^y\} = \widetilde{F}^x$, hence $\gamma'_\sigma(\widetilde{F}^y, A(F)) = \sigma(F)$ for each $F \in \mathscr{A}$. □

In order to prove unverifiability of a semantics σ w.r.t. a class induced by a certain $\mathfrak{r}^x$ it suffices to present two AFs F and G such that $\sigma(F) \neq \sigma(G)$ but, $\widetilde{F}^x = \widetilde{G}^x$ and $A(F) = A(G)$. Then the verification class induced by $\mathfrak{r}^x$ does not provide enough information to verify σ.

In the following we will use this strategy to show exact verifiability. Consider a semantics σ which is verifiable by a class induced by $\mathfrak{r}^x$. If σ is unverifiable by all verifiability classes induced by $\mathfrak{r}^y$ with $\mathfrak{r}^y \prec \mathfrak{r}^x$ we have that σ is exactly verifiable by $\mathfrak{r}^x$. The following examples study this issue for the semantics under consideration.

Example 4. The complete semantics is $+-$-verifiable as seen before. The following AFs show that it is even exactly verifiable by that class.



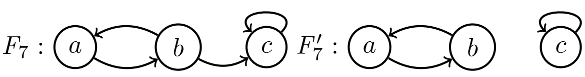
First consider the AFs F_1 and F'_1, and observe that $\widetilde{F_1}^{+\pm} = \{(\emptyset, \emptyset, \emptyset), (\{a\}, \emptyset, \emptyset)\} = \widetilde{F'_1}^{+\pm}$. On the other hand F_1 and F'_1 differ in their complete extensions since $co(F_1) = \{\emptyset\}$ but $co(F'_1) = \{\{a\}\}$. Therefore complete semantics is unverifiable by the verification class induced by $\mathfrak{r}^{+\pm}$. Likewise, this can be shown for the classes induced by $\mathfrak{r}^{-\mp}$, $\mathfrak{r}^{\pm\mp}$, $\mathfrak{r}^{-\pm}$, $\mathfrak{r}^{+\mp}$, and $\mathfrak{r}^{\cap\cup}$, respectively:

- $\widetilde{F_2}^{-\mp} = \{(\emptyset, \emptyset, \emptyset), (\{a\}, \emptyset, \emptyset), (\{a, c\}, \{b\}, \emptyset), (\{c\}, \{b\}, \emptyset)\} = \widetilde{F'_2}^{-\mp}$, but $co(F_2) = \{\{a\}, \{a, c\}\} \neq \{\{a, c\}\} = co(F'_2)$.
- $\widetilde{F_3}^{\pm\mp} = \widetilde{F'_3}^{\pm\mp}$, but $co(F_3) = \{\emptyset, \{a\}\} \neq \{\{a\}\} = co(F'_3)$.
- $\widetilde{F_4}^{-\pm} = \widetilde{F'_4}^{-\pm}$, but $co(F_4) = \{\emptyset, \{a\}\} \neq \{\emptyset\} = co(F'_4)$.
- $\widetilde{F_5}^{+\mp} = \widetilde{F'_5}^{+\mp}$, but $co(F_5) = \{\emptyset, \{a\}\} \neq \{\{a\}\} = co(F'_5)$.
- $\widetilde{F_6}^{\cap\cup} = \widetilde{F'_6}^{\cap\cup}$, but $co(F_6) = \{\{a\}\} \neq \{\emptyset\} = co(F'_6)$.

Hence the complete semantics is exactly verifiable by the verification class induced by $\mathfrak{r}^{+-}$.

Example 5. Consider the semi-stable and eager semantics and recall that they are $+\mp$-verifiable In order to show exact verifiability it suffices to show unverifiability by the classes induced by $\mathfrak{r}^{+}$, $\mathfrak{r}^{\cup}$, and $\mathfrak{r}^{\mp}$ (cf. Figure 1); F_1 and F_6 are taken from Example 4 above.

- $\widetilde{F_1}^{+} = \widetilde{F'_1}^{+}$, but $ss(F_1) = eg(F_1) = \{\emptyset\} \neq \{\{a\}\} = ss(F'_1) = eg(F'_1)$.
- $\widetilde{F_6}^{\cup} = \widetilde{F'_6}^{\cup}$, but $ss(F_6) = eg(F_6) = \{\{a\}\} \neq \{\emptyset\} = ss(F'_6) = eg(F'_6)$.
- $\widetilde{F_7}^{\mp} = \widetilde{F'_7}^{\mp}$, but $ss(F_7) = \{\{b\}\} \neq \{\{a\}, \{b\}\} = ss(F'_7)$ and $eg(F_7) = \{\{b\}\} \neq \{\emptyset\} = eg(F'_7)$.



Hence, both the semi-stable and eager semantics are exactly verifiable by the verification class induced by $\mathfrak{r}^{+\mp}$.

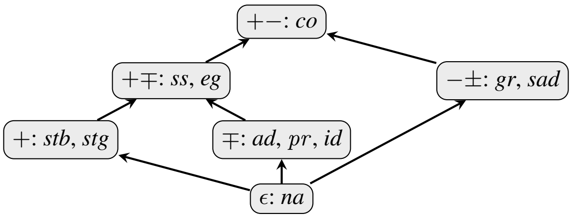


Figure 2: Semantics and their exact verification classes.

Example 6. Now consider the grounded and strong admissible semantics and recall that they are $-\pm$-verifiable In order to show exact verifiability we have to show unverifiability by the classes induced by $\mathfrak{r}^{\pm}$, $\mathfrak{r}^{-}$, and $\mathfrak{r}^{\cup}$ (cf. Figure 1); again, the AFs from Example 4 can be reused.

- $\widetilde{F_1}^{\pm} = \widetilde{F_1'}^{\pm}$, but $gr(F_1) = \{\emptyset\} \neq \{\{a\}\} = gr(F_1')$ and $sad(F_1) = \{\emptyset\} \neq \{\emptyset, \{a\}\} = sad(F_1')$.
- $\widetilde{F_2}^{-} = \widetilde{F_2'}^{-}$, but $gr(F_2) = \{\{a\}\} \neq \{a, c\} = gr(F_2')$ and $sad(F_2) = \{\emptyset, \{a\}\} \neq \{\emptyset, \{a\}, \{a, c\}\} = sad(F_2')$
- $\widetilde{F_6}^{\cup} = \widetilde{F_6'}^{\cup}$, but $gr(F_6) = \{\{a\}\} \neq \{\emptyset\} = gr(F_6')$ and $sad(F_6) = \{\emptyset, \{a\}\} \neq \{\emptyset\} = sad(F_6')$.

Hence, both the grounded and strong admissible semantics are exactly verifiable by the verification class induced by $\mathfrak{r}^{+\mp}$.

Example 7. Finally consider stable, stage, admissible, preferred and ideal semantics. They are either $+$-verifiable (*stb* and stg) or $\mp$-verifiable (ad, pr, and id). In order to show that these verification classes are exact we have to show unverifiability w.r.t. the verification class induced by $\mathfrak{r}^{\epsilon}$. Consider, for instance, the AFs F_4 and F_4' from Example 4. We have $\widetilde{F_4}^{\epsilon} = \widetilde{F_4'}^{\epsilon}$, but $ad(F_4) = \{\emptyset, \{a\}\} \neq \{\emptyset\} = ad(F_4')$, $stb(F_4) = \{\{a\}\} \neq \emptyset = stb(F_4')$, and $\sigma(F_4) = \{\{a\}\} \neq \{\emptyset\} = \sigma(F_4')$ for $\sigma \in \{stg, pr, id\}$, showing exactness of the respective verification classes.

The insights obtained through Examples 4, 5, 6, and 7 show that the verification classes obtained from the criteria given above are indeed exact. Figure 2 shows the relation between the semantics under consideration with respect to their exact verification classes.

We turn now to the main theorem stating that any *rational* semantics (recall that all semantics we consider in this paper are rational) is exactly verifiable by one of the 15 different verification classes.

Theorem 4. *Every semantics which is rational is exactly verifiable by a verification class induced by one of the neighborhood functions presented in Figure 1.*

Proof. First of all note that by Lemma 2, $\mathfrak{r}^{\epsilon}$ is the least informative neighborhood function and for every other neighborhood function $\mathfrak{r}^{x}$ it holds that $\mathfrak{r}^{\epsilon} \preceq \mathfrak{r}^{-}$. Therefore, if a semantics is verifiable by the verification class induced by any $\mathfrak{r}^{x}$ then it is exactly verifiable by a verification class induced by some $\mathfrak{r}^{y}$ with $\mathfrak{r}^{\epsilon} \preceq \mathfrak{r}^{y} \preceq \mathfrak{r}^{x}$. Moreover, if a semantics is exactly verifiable by a class, then it is by definition also verifiable by this class. Hence it remains to show that every semantics which is rational is verifiable by a verification class presented in Figure 1.

We show the contrapositive, i.e., if a semantics is not verifiable by a verification class induced by one of the neighborhood functions presented in Figure 1 then it is not rational.

Assume a semantics σ is not verifiable by one of the verification classes. This means σ is not verifiable by the verification class induced by $\mathfrak{r}^{+-}$. Hence there exist two AFs F and G such that $\widetilde{F}^{+-} = \widetilde{G}^{+-}$ and $A(F) = A(G)$, but $\sigma(F) \neq \sigma(G)$. For every argument a which is not self-attacking, a tuple $(\{a\}, \{a\}^{+}, \{a\}^{-})$ is contained in $\widetilde{F}^{+-}$ (and in $\widetilde{G}^{+-}$). Hence F and G have the same not-self-attacking arguments and, moreover these arguments have the same ingoing and outgoing attacks in F and G. This, together with $A(F) = A(G)$ implies that $F^{l} = G^{l}$ (see Definition 2) holds. But since $\sigma(F) \neq \sigma(G)$ we get that σ is not rational, which was to show. □

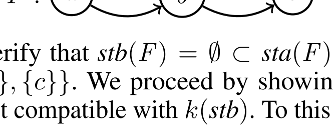
Note that the criterion giving evidence for verifiability of a semantics by a certain class has access to the set of arguments of a given framework. In fact, only the criterion for stable semantics makes use of that. Indeed, stable semantics needs this information since it is not verifiable by any class when using a weaker notion of verifiability, which rules out the usage of $A(F)$.

Intermediate Semantics

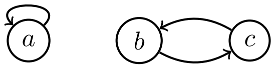
A type of semantics which has aroused quite some interest in the literature (see e.g. (Baroni and Giacomin 2007a) and (Nieves, Osorio, and Zepeda 2011)) are intermediate semantics, i.e. semantics which yield results lying between two existing semantics. The introduction of σ-τ-intermediate semantics can be motivated by deleting *undesired* (or add *desired*) τ-extensions while guaranteeing all reasonable positions w.r.t. σ. In other words, σ-τ-intermediate semantics can be seen as sceptical or credulous acceptance shifts within the range of σ and τ.

A natural question is whether we can make any statements about compatible kernels of intermediate semantics. In particular, if semantics σ and τ are compatible with some kernel k, is then every σ-τ-intermediate semantics k-compatible. The following example answers this question negatively.

Example 8. Recall from Theorem 1 that both stable and stage semantics are compatible with $k(stb)$, i.e. $F \equiv_E^{stb} G \Leftrightarrow F \equiv_E^{stg} G \Leftrightarrow F^{k(stb)} = G^{k(stb)}$. Now we define the following *stb*-*stg*-intermediate semantics, say *stagle* semantics: Given an AF $F = (A, R)$, $S \in sta(F)$ iff $S \in cf(F)$, $S_F^{+} \cup S_F^{-} = A$ and for every $T \in cf(F)$ we have $S_F^{+} \not\subset T_F^{+}$. Obviously, it holds that $stb \subseteq sta \subseteq stg$ and $stb \neq sta$ as well as $sta \neq stg$, as witnessed by the following AF F:



It is easy to verify that $stb(F) = \emptyset \subset sta(F) = \{\{b\}\} \subset stg(F) = \{\{b\}, \{c\}\}$. We proceed by showing that stagle semantics is not compatible with $k(stb)$. To this end consider $F^{k(stb)}$, which is depicted below.

$F^{k(stb)}$:


Now, $sta\left(F^{k(stb)}\right) = \{\{b\},\{c\}\}$ witnesses $F \not\equiv^{sta} F^{k(stb)}$ and therefore, $F \not\equiv^{sta}_E F^{k(stb)}$. Since $F^{k(stb)} = \left(F^{k(stb)}\right)^{k(stb)}$ we are done, i.e. stagle semantics is indeed not compatible with the stable kernel.

It is the main result of this section that compatibility of intermediate semantics w.r.t. a certain kernel can be guaranteed if verifiability w.r.t. a certain class is presumed. The provided characterization theorems generalize former results presented in (Oikarinen and Woltran 2011). Moreover, due to the abstract character of the theorems the results are applicable to semantics which may be defined in the future.

Before turning to the characterization theorems we state some implications of verifiability. In particular, under the assumption that σ is verifiable by a certain class, equality of certain kernels implies expansion equivalence w.r.t. σ.

Proposition 4. *For any $+$-verifiable semantics σ we have*

$$F^{k(stb)} = G^{k(stb)} \Rightarrow F \equiv^{\sigma}_{E} G.$$

Proof. In (Oikarinen and Woltran 2011) it was shown that $F^{k(stb)} = G^{k(stb)} \Rightarrow (F \cup H)^{k(stb)} = (G \cup H)^{k(stb)}$ (i). Consider now a $+$-verifiable semantics σ. In order to show $\sigma(F) = \sigma\left(F^{k(stb)}\right)$ (ii) we prove $\widetilde{F}^{+} = \widetilde{F^{k(stb)}}^{+}$ (*) first. It is easy to see that $S \in cf(F)$ iff $S \in cf\left(F^{k(stb)}\right)$. Furthermore, since $k(stb)$ deletes an attack (a,b) only if a is self-defeating we deduce that ranges does not change as long as conflict-free sets are considered. Thus, $\sigma(F) =_{(\text{Def.})} \gamma_\sigma(\widetilde{F}^{+}) =_{(*)} \gamma_\sigma(\widetilde{F^{k(stb)}}^{+}) =_{(\text{Def.})} \sigma(F^{k(stb)})$.

Now assume that $F^{k(stb)} = G^{k(stb)}$ and let $S \in \sigma(F \cup H)$ for some AF H. We have to show that $S \in \sigma(G \cup H)$. Applying (ii) we obtain $S \in \sigma\left((F \cup H)^{k(stb)}\right)$. Furthermore, using (i) we deduce $S \in \sigma\left((G \cup H)^{k(stb)}\right)$. Finally, $S \in \sigma(G \cup H)$ by applying (ii), which concludes the proof. □

The following results can be shown in a similar manner.

Proposition 5. *For any $+\mp$-verifiable semantics σ we have*

$$F^{k(ad)} = G^{k(ad)} \Rightarrow F \equiv^{\sigma}_{E} G.$$

Proposition 6. *For any $+-$-verifiable semantics σ we have*

$$F^{k(co)} = G^{k(co)} \Rightarrow F \equiv^{\sigma}_{E} G.$$

Proposition 7. *For any $-\pm$-verifiable semantics σ we have*

$$F^{k(gr)} = G^{k(gr)} \Rightarrow F \equiv^{\sigma}_{E} G.$$

Proposition 8. *For any ϵ-verifiable semantics σ we have*

$$F^{k(na)} = G^{k(na)} \Rightarrow F \equiv^{\sigma}_{E} G.$$

We proceed with general characterization theorems. The first one states that *stb*-*stg*-intermediate semantics are compatible with stable kernel if $+$-verifiability is given. Consequently, stagle semantics as defined in Example 8 can not be $+$-verifiable.

Theorem 5. *Given a semantics σ which is $+$-verifiable and stb-stg-intermediate, it holds that*

$$F^{k(stb)} = G^{k(stb)} \Leftrightarrow F \equiv^{\sigma}_{E} G.$$

Proof. ($\Rightarrow$) Follows directly from Proposition 4.
($\Leftarrow$) We show the contrapositive, i.e. $F^{k(stb)} \neq G^{k(stb)} \Rightarrow F \not\equiv^{\sigma}_{E} G$. Assuming $F^{k(stb)} \neq G^{k(stb)}$ implies $F \not\equiv^{stg}_{E} G$, i.e. there exists an AF H such that $stg(F \cup H) \neq stg(G \cup H)$ and therefore, $stb(F \cup H) \neq stb(G \cup H)$. Let $B = A(F) \cup A(G) \cup A(H)$ and $H' = (B \cup \{a\}, \{(a,b),(b,a) \mid b \in B\})$. It is easy to see that $stb(F \cup H') = stb(F \cup H) \cup \{\{a\}\}$ and $stb(G \cup H') = stb(G \cup H) \cup \{\{a\}\}$. Since now both $stb(F \cup H') \neq \emptyset$ and $stb(G \cup H') \neq \emptyset$ it holds that $stb(F \cup H') = stg(F \cup H')$ and $stb(G \cup H') = stg(G \cup H')$. Hence $\sigma(F \cup H') \neq \sigma(F \cup H')$, showing that $F \not\equiv^{stb}_{E} G$. □

The following theorems can be shown in a similar manner.

Theorem 6. *Given a semantics σ which is $+\mp$-verifiable and ρ-ad-intermediate with $\rho \in \{ss, id, eg\}$, it holds that*

$$F^{k(ad)} = G^{k(ad)} \Leftrightarrow F \equiv^{\sigma}_{E} G.$$

Remember that complete semantics is a *ss*-*ad*-intermediate semantics. Furthermore, it is not characterizable by the admissible kernel as already observed in (Oikarinen and Woltran 2011). Consequently, complete semantics is not $+\mp$-verifiable (as we have shown in Example 4 with considerable effort).

Theorem 7. *Given a semantics σ which is $-\pm$-verifiable and gr-sad-intermediate, it holds that*

$$F^{k(gr)} = G^{k(gr)} \Leftrightarrow F \equiv^{\sigma}_{E} G.$$

Conclusions

In this work we have contributed to the analysis and comparison of abstract argumentation semantics. The main idea of our approach is to provide a novel categorization in terms of the amount of information required for testing whether a set of arguments is an extension of a certain semantics. The resulting notion of verifiability classes allows us to categorize any new semantics (given it is "rational") with respect to the information needed and compare it to other semantics. Thus our work is in the tradition of the principle-based evaluation due to Baroni and Giacomin (2007b) and paves the way for a more general view on argumentation semantics, their common features, and their inherent differences.

Using our notion of verifiability, we were able to show kernel-compatibility for certain intermediate semantics. Concerning concrete semantics, our results yield the following observation: While preferred, semi-stable, ideal and eager semantics coincide w.r.t. strong equivalence, verifiability of these semantics differs. In fact, preferred and ideal semantics manage to be verifiable with strictly less information.

For future work we envisage an extension of the notion of verifiability classes in order to categorize semantics not captured by the approach followed in this paper, such as *cf2* (Baroni, Giacomin, and Guida 2005).

References

Arieli, O. 2012. Conflict-tolerant semantics for argumentation frameworks. In *Logics in Artificial Intelligence - 13th European Conference, Proceedings*, volume 7519 of *Lecture Notes in Computer Science*, 28–40. Springer.

Baroni, P., and Giacomin, M. 2007a. Comparing argumentation semantics with respect to skepticism. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 9th European Conference, Proceedings*, volume 4724 of *Lecture Notes in Computer Science*, 210–221. Springer.

Baroni, P., and Giacomin, M. 2007b. On principle-based evaluation of extension-based argumentation semantics. *Artif. Intell.* 171(10-15):675–700.

Baroni, P.; Caminada, M.; and Giacomin, M. 2011. An introduction to argumentation semantics. *Knowledge Eng. Review* 26(4):365–410.

Baroni, P.; Dunne, P. E.; and Giacomin, M. 2011. On the resolution-based family of abstract argumentation semantics and its grounded instance. *Artif. Intell.* 175(3-4):791–813.

Baroni, P.; Giacomin, M.; and Guida, G. 2005. SCC-Recursiveness: A general schema for argumentation semantics. *Artif. Intell.* 168(1-2):162–210.

Baumann, R., and Brewka, G. 2013. Analyzing the equivalence zoo in abstract argumentation. In *14th International Workshop on Computational Logic in Multi-Agent Systems, Proceedings*, volume 8143 of *Lecture Notes in Computer Science*, 18–33. Springer.

Baumann, R., and Brewka, G. 2015. The equivalence zoo for Dung-style semantics. *Journal of Logic and Computation.*

Baumann, R., and Spanring, C. 2015. Infinite argumentation frameworks – on the existence and uniqueness of extensions. In *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation - Essays Dedicated to G. Brewka on the Occ. of His 60th Birthday*, volume 9060 of *Lecture Notes in Computer Science*, 281–295. Springer.

Baumann, R., and Woltran, S. 2014. The role of self-attacking arguments in characterizations of equivalence notions. *Journal of Logic and Computation: Special Issue on Loops in Argumentation.*

Baumann, R. 2016. Characterizing equivalence notions for labelling-based semantics. In *Principles of Knowledge Representation and Reasoning: Proceedings of the 15th International Conference*. To appear.

Besnard, P.; Garcia, A.; Hunter, A.; Modgil, S.; Prakken, H.; Simari, G.; and Toni, F. 2014. Special issue: Tutorials on structured argumentation. *Argument and Computation* 5(1):1–117.

Caminada, M.; Carnielli, W. A.; and Dunne, P. E. 2012. Semi-stable semantics. *J. Log. Comput.* 22(5):1207–1254.

Caminada, M. 2007. Comparing two unique extension semantics for formal argumentation: Ideal and eager. In *19th Belgian-Dutch Conference on Artificial Intelligence, Proceedings*, 81–87.

Caminada, M. 2014. Strong admissibility revisited. In *Computational Models of Argument - Proceedings of COMMA 2014*, volume 266 of *Frontiers in Artificial Intelligence and Applications*, 197–208. IOS Press.

Dung, P. M.; Mancarella, P.; and Toni, F. 2007. Computing ideal sceptical argumentation. *Artif. Intell.* 171(10-15):642–674.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* 77(2):321–357.

Dvořák, W.; Linsbichler, T.; Oikarinen, E.; and Woltran, S. 2014. Resolution-based grounded semantics revisited. In *Computational Models of Argument - Proceedings of COMMA 2014*, volume 266 of *Frontiers in Artificial Intelligence and Applications*, 269–280. IOS Press.

Gaggl, S. A., and Woltran, S. 2013. The cf2 argumentation semantics revisited. *J. Log. Comput.* 23(5):925–949.

Grossi, D., and Modgil, S. 2015. On the graded acceptability of arguments. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 868–874. AAAI Press.

Jakobovits, H., and Vermeir, D. 1999. Robust semantics for argumentation frameworks. *J. Log. Comput.* 9(2):215–261.

Lifschitz, V.; Pearce, D.; and Valverde, A. 2001. Strongly equivalent logic programs. *ACM Transactions on Computational Logic* 2(4):526–541.

Loui, R. P. 1987. Defeat among arguments: a system of defeasible inference. *Computational Intelligence* 14(1):100–106.

Maher, M. J. 1986. Eqivalences of logic programs. In *3rd International Conference on Logic Programming, Proceedings*, volume 225 of *Lecture Notes in Computer Science*, 410–424. Springer.

Nieves, J. C.; Osorio, M.; and Zepeda, C. 2011. A schema for generating relevant logic programming semantics and its applications in argumentation theory. *Fundam. Inform.* 106(2-4):295–319.

Oikarinen, E., and Woltran, S. 2011. Characterizing strong equivalence for argumentation frameworks. *Artif. Intell.* 175(14-15):1985–2009.

Pollock, J. L. 1987. Defeasible reasoning. *Cognitive Science* 11(4):481–518.

Prakken, H., and Vreeswijk, G. 2002. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*. Dordrecht. 219–318.

Truszczynski, M. 2006. Strong and uniform equivalence of nonmonotonic theories - an algebraic approach. *Annals of Mathematics and Artificial Intelligence* 48(3-4):245–265.

Turner, H. 2004. Strong equivalence for causal theories. In *7th International Conference on Logic Programming and Nonmonotonic Reasoning, Proceedings*, volume 2923 of *Lecture Notes in Computer Science*, 289–301. Springer.

Verheij, B. 1996. Two approaches to dialectical argumentation: admissible sets and argumentation stages. In *8th Dutch Conference on Artificial Intelligence, Proceedings*, 357–368.

Preferential Modalities Revisited

Katarina Britz
CSIR-SU Centre for AI Research
Stellenbosch University, South Africa
abritz@sun.ac.za

Ivan Varzinczak
Centre de Recherche en Informatique de Lens
Université d'Artois, France
varzinczak@cril.fr

Abstract

We venture beyond the customary semantic approach in NMR, namely that of placing orderings on worlds (or valuations). In a modal-logic setting, we motivate and investigate the idea of ordering elements of the accessibility relations in Kripke frames, i.e., world pairs (w, w') (or 'arrows'). The underlying intuition is that some world pairs may be seen as more normal (or typical, or expected) than others. We show this delivers an elegant and intuitive semantic construction, which gives a new perspective on present notions of defeasible necessity. From a modeler's perspective, the new framework we propose is more intuitively appealing. Technically, though, the revisited logic happens to not substantively increase the expressive power of the previously defined preferential modalities. This conclusion follows from an analysis of both semantic constructions via a generalisation of bisimulations to the preferential case. Lest this be seen as a negative result, it essentially means that reasoners based on the previous semantics (which have been shown to preserve the computational complexity of the underlying classical modal language) suffice for reasoning over the new semantics. Finally, we show that the kind of construction we here propose has many fruitful applications, notably in a description-logic context, where it provides the foundations on which to ground useful notions of defeasibility in ontologies yet to be explored.

Introduction

Accounts of normality (or typicality), plausibility and alike traditionally have an underlying semantics built on a notion of preference on *worlds*. Such is the case of non-monotonic entailment (Shoham 1988; Kraus, Lehmann, and Magidor 1990; Makinson 2005), conditionals (Lehmann and Magidor 1992; Boutilier 1994), belief revision (Katsuno and Mendelzon 1991; Baltag and Smets 2006; 2008), counterfactuals (Stalnaker 1968; Lewis 1973; 1974), obligations (Hansson 1969) and many others, as known from the literature on non-monotonic reasoning, conditional and deontic logics, and related areas. Roughly speaking, the usual approach consists in selecting some worlds (or propositional valuations) as being more normal (alias typical, alias desirable) and carrying out the reasoning relative to an underlying normality ordering on worlds.

A typical representative of these different yet interrelated threads of investigation is the well-known preferential approach (Shoham 1988) and its derivatives (Kraus, Lehmann, and Magidor 1990; Lehmann and Magidor 1992). There, a preference relation is defined on the set of possible worlds with the (tacit) assumption that these contain all is needed to reason about what is normal or expected. A case can indeed be made for such an assumption in a propositional setting. However, in logics with more structure, it is reasonable to say that the normality 'spotlight' should not be confined to worlds, but rather (also) be put on (possibly) whatever structure one has at one's disposal in the respective underlying semantics. To witness, in a modal logic context, it makes sense to ask whether some links between worlds in a frame are (relatively) more normal (or preferred) than others—irrespective of whether the worlds involved are by any means comparable in that way or another amongst themselves. In other words, one can be interested in the normality of the *transition* from one world to another one. This point is better illustrated with some well-known concrete applications of modal logics as given below.

Let us assume a very simplistic scenario in which we have only one propositional atom, on, of which the intuition is that a particular light-bulb is on. Moreover, let us assume there is only one action at one's disposal, namely toggle (hereafter abbreviated t), of which the intuition is that of changing the state of the light switch. Figure 1 below depicts a possible-worlds model for this scenario.

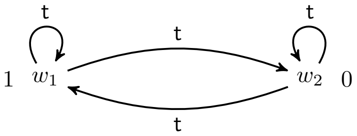


Figure 1: A possible-worlds model for one action (toggle) and one atom (on).

Intuitively, normal (or typical, or expected) executions of the toggle action are given by the t-transitions from w_1 to w_2 and back, whereas the reflexive arrows are, in a sense, less expected in the given scenario. Therefore, it becomes important to single out those executions of the action that are deemed more normal from those that are not. In Figure 1, this would amount to enriching the semantic structure (in a way still to be defined) with information specifying that the pairs (w_1, w_2) and (w_2, w_1) somehow take 'precedence' over (w_1, w_1) and (w_2, w_2) when reasoning about possible executions of the action.

Let us now consider a variant of the above scenario (although just as simple), in which we have only one atomic proposition, correct (which we shall abbreviate as c), the intuition of which is that a proposed proof for a mathematical statement is correct. Furthermore, let us assume there is a good mathematician, M, whose knowledge about the correctness of the proof is of interest to us. Figure 2 below depicts one possible configuration of this scenario.

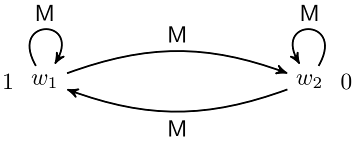


Figure 2: A possible-worlds model for one agent (the good mathematician M) and one atom (correct).

As a good mathematician, Agent M should know whether the proof is correct. Nevertheless, in the model of Figure 2, M does not know (in the classical sense) whether the proof is correct or not, since, also by virtue of being a good mathematician, M admits the (unlikely) possibility of being wrong (at least until the proof has been submitted to peer-reviewed scrutiny). In this case, we would say that Agent M *defeasibly knows* whether the proof is correct or not, an epistemic stance that can be adopted by 'focusing' on the most normal (or expected) of the epistemic possibilities held by the agent, namely (w_1, w_1) and (w_2, w_2) in Figure 2, which, in this example, are more normal than (w_1, w_2) and (w_2, w_1). (It is not hard to see that the motivation above also holds in a doxastic context, as certain beliefs may be more entrenched than others.)

In order to motivate the foregoing ideas in a deontic context, let us assume a language with a single propositional atom, namely fair-play, henceforth abbreviated f and of which the intuition is that, in a competition, the players abide by an established standard of 'decency' or an 'honorable conduct'. In this context, adopting a fair-play stance is not to be seen as an obligation in the usual (strict) meaning of the term. It is rather a matter of best practice in that it corresponds to the expected, though not enforceable (even if, in some cases, liability-biding), attitude. Figure 3 below depicts a possible-worlds model for this scenario.

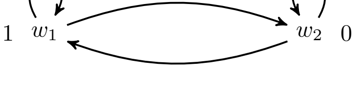


Figure 3: A possible-worlds model for one atom (fair-play) in a deontic-context.

A case can be made that envisioning an f-world as a better alternative (to the current one) is more appropriate than the contemplation of a ¬f-one. Semantically, this requirement would be translated as setting the pairs (w_1, w_1) and (w_2, w_1) as more preferred than (w_2, w_2) and (w_1, w_2). In this specific example, it happens that we could also model the underlying preference as an ordering on worlds, with f-worlds preferred to the ¬f-worlds. However, in the preceding examples, ordering worlds rather than pairs of worlds is neither intuitive nor is it immediately clear whether this is even possible.

In this work, we address precisely these issues. We shall start by shifting the normality spotlight from possible worlds to transitions amongst them, i.e., to accessibility relations in Kripke frames. The justification for doing so stems from a comparison with the classical (monotonic) case: In classical Kripkean semantics, modalities are primarily about accessibility, only secondarily about worlds' contents. Hence, we contend that accounts of a notion of defeasibility in modalities (like those illustrated above) should primarily focus on normality of the accessibility relations rather than (or at least prior to) that of the (accessible) worlds. With that we hope to pave the way for further explorations of non-monotonicity in modal logics, in particular in extensions of the preferential approach therein (Britz, Meyer, and Varzinczak 2011a).

Preliminaries

In this section, we provide the required formal background for the rest of this work. In particular, we set up the notation and conventions that shall be followed in the upcoming sections. (The reader conversant with modal logic can safely skip the first subsection below.)

Modal Logic

We work in a set of *atomic propositions* $\mathcal{P}$, using the logical connectives $\wedge$ (conjunction), $\neg$ (negation), and a set of modal operators $\Box_i$, $1 \leqslant i \leqslant n$. Propositions are denoted by $p, q, \ldots$, and sentences by $\alpha, \beta, \ldots$, constructed in the usual way according to the rule ($1 \leqslant i \leqslant n$):

$$\alpha ::= p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \Box_i \alpha$$

All the other truth-functional connectives ($\vee$, $\rightarrow$, $\leftrightarrow$, ...) are defined in terms of $\neg$ and $\wedge$ in the usual way. Given $\Box_i$, $1 \leqslant i \leqslant n$, with $\Diamond_i$ we denote its *dual* modal operator, i.e., for any α, $\Diamond_i \alpha := \neg\Box_i\neg\alpha$. We use $\top$ as an abbreviation for $p \vee \neg p$ and $\bot$ as an abbreviation for $p \wedge \neg p$, for some $p \in \mathcal{P}$. With $\mathcal{L}^{\Box}$ we denote the set of all sentences of the modal language.

The semantics is the standard possible-worlds one:

Definition 1 (Kripke Model) *A Kripke model is a tuple $\mathscr{M} := \langle W, R, V \rangle$ where W is a (non-empty) set of possible worlds, $R := \langle R_1, \ldots, R_n \rangle$, where each $R_i \subseteq W \times W$ is an accessibility relation on W, $1 \leqslant i \leqslant n$, and $V : W \longrightarrow \{0, 1\}^{\mathcal{P}}$ is a valuation function mapping possible worlds into propositional valuations.*

As an example, Figure 4 depicts the Kripke model $\mathscr{M}_1 = \langle W_1, R_1, V_1 \rangle$, where $W_1 := \{w_i \mid 1 \leqslant i \leqslant 4\}$, $R_1 := \langle R_a, R_b \rangle$, with $R_a := \{(w_1, w_2), (w_1, w_3), (w_4, w_3)\}$, and $R_b := \{(w_1, w_4), (w_2, w_3)\}$, and V_1 is the obvious valuation function.

In our pictorial representations of models, we represent propositional valuations as sequences of 0s and 1s, and with the obvious implicit ordering of atoms. Thus, for the logic generated from p and q, the valuation in which p is true and q is false will be represented as 10.

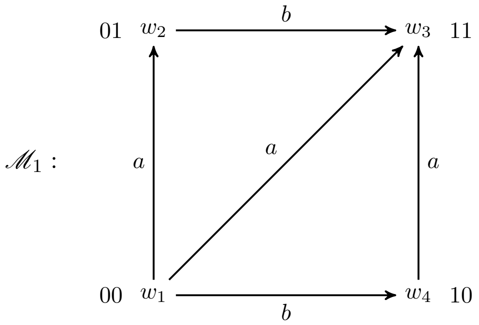


Figure 4: A Kripke model for $\mathcal{P} = \{p, q\}$ and two modalities, namely a and b.

We shall use $w, u, v, \ldots$ (possibly decorated with primes) to denote possible worlds. Moreover, where it aids readability, we shall henceforth sometimes write tuples of the form (w, w') as ww'.

Sentences of $\mathcal{L}^{\Box}$ are true or false relative to a possible world in a given Kripke model:

Definition 2 (Truth Conditions) *Let $\mathscr{M} = \langle W, R, V \rangle$ and $w \in W$:*

- *$\mathscr{M}, w \Vdash p$ if and only if $V(w)(p) = 1$;*
- *$\mathscr{M}, w \Vdash \neg\alpha$ if and only if $\mathscr{M}, w \nVdash \alpha$;*
- *$\mathscr{M}, w \Vdash \alpha \wedge \beta$ if and only if $\mathscr{M}, w \Vdash \alpha$ and $\mathscr{M}, w \Vdash \beta$;*
- *$\mathscr{M}, w \Vdash \Box_i\alpha$ if and only if $\mathscr{M}, w' \Vdash \alpha$ for all w' such that $(w, w') \in R_i$.*

Given $\alpha \in \mathcal{L}^{\Box}$ and $\mathscr{M} = \langle W, R, V \rangle$, we say that $\mathscr{M}$ *satisfies* α if there is at least one world $w \in W$ such that $\mathscr{M}, w \Vdash \alpha$. We say that $\mathscr{M}$ is a *model of* α (alias α is *true* in $\mathscr{M}$), denoted $\mathscr{M} \Vdash \alpha$, if $\mathscr{M}, w \Vdash \alpha$ for every world $w \in W$. Given a class (i.e., a collection) of models $\mathcal{M}$, we say that α is *valid* in $\mathcal{M}$, denoted $\models_{\mathcal{M}} \alpha$, if and only if every Kripke model $\mathscr{M} \in \mathcal{M}$ is a model of α. Given $\mathcal{K} \subseteq \mathcal{L}^{\Box}$ and $\alpha \in \mathcal{L}^{\Box}$, we say that $\mathcal{K}$ *locally entails* α in the class of models $\mathcal{M}$, denoted $\mathcal{K} \models_{\mathcal{M}} \alpha$, if and only if for every Kripke model $\mathscr{M} \in \mathcal{M}$ and every w in $\mathscr{M}$, if $\mathscr{M}, w \Vdash \beta$ for every $\beta \in \mathcal{K}$, then $\mathscr{M}, w \Vdash \alpha$. (When the class of models we are working with is clear from the context, we shall dispense with subscripts and just write $\models \alpha$ and $\mathcal{K} \models \alpha$.)

Here we shall assume the system of normal modal logic K, of which all the other normal modal logics are extensions. Semantically, K is characterised by the class of all Kripke models (Definition 1). Syntactically, K corresponds to the smallest set of sentences containing all propositional tautologies, all instances of the axiom schema $\mathsf{K} : \Box_i(\alpha \to \beta) \to (\Box_i\alpha \to \Box_i\beta)$, $1 \leqslant i \leqslant n$, and closed under the *rule of necessitation* below:

$$\text{(RN)}\ \frac{\alpha}{\Box_i\alpha} \tag{1}$$

For more details on modal logic, we refer the reader to the handbook by Blackburn *et al.* (2006).

Preferential Modalities

In previous work (Britz, Meyer, and Varzinczak 2011a; Britz and Varzinczak 2013), we have investigated the fruitfulness of extending the standard Kripke semantics with a preference relation on the set of possible worlds. This gives rise to the following semantic structure, of which the underlying motivation is similar to that behind Boutilier's (1994) CT4O models and the plausibility models of Baltag and Smets (2006; 2008).

Definition 3 (W-Ordered Model) *A W-ordered model is a tuple $\mathscr{W} := \langle W, R, V, \prec \rangle$ where $\langle W, R, V \rangle$ is as in Definition 1 and $\prec\ \subseteq W \times W$ is a well-founded strict partial order on W, i.e., $\prec$ is irreflexive, transitive and every non-empty $W' \subseteq W$ has minimal elements w.r.t. $\prec$ (see Definition 4).*

The intuition behind the preference relation $\prec$ in a W-ordered model $\mathscr{W}$ is that the worlds lower down in the ordering are deemed as more preferred (or more normal) than those higher up.

Definition 4 (Minimality w.r.t. $\prec$) *Let $\mathscr{W} = \langle W, R, V, \prec \rangle$ be a W-ordered model and let $X \subseteq W$. Then $\min_{\prec} X := \{w \in X \mid$ there is no $w' \in X$ such that $w' \prec w\}$, i.e., $\min_{\prec} X$ denotes the minimal elements of X with respect to the preference relation $\prec$.*

As an example, Figure 5 below depicts the W-ordered model $\mathscr{W}_1 = \langle W_1, R_1, V_1, \prec_1 \rangle$, where $\langle W_1, R_1, V_1 \rangle$ is as in Figure 4 and $\prec_1\ := \{(w_1, w_2), (w_2, w_3), (w_1, w_3), (w_4, w_3)\}$.

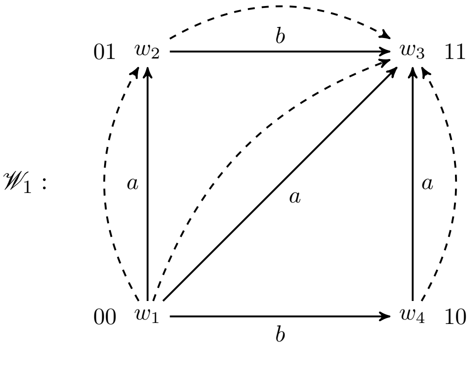


Figure 5: A W-ordered model for $\mathcal{P} = \{p, q\}$ and two modalities (a and b). The preference relation $\prec_1$ is represented by the dashed arrows, which point from more preferred to less preferred worlds.

We can then extend $\mathcal{L}^{\Box}$ with a family of defeasible modal operators ⚐$_i$ (called 'flag'), $1 \leqslant i \leqslant n$, where n is the number of classical modalities in the language. The sentences of the extended language are then recursively defined by:

$$\alpha ::= p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \Box_i\alpha \mid \Bumpeq_i\alpha$$

As before, the other connectives are defined in terms of $\neg$ and $\wedge$ in the usual way, $\top$ and $\bot$ are seen as abbreviations, and $\Diamond_i$ is the dual of $\Box_i$. Moreover, with $\lozenge_i$ (called 'flame') we denote the dual of $\Bumpeq_i$. We shall use $\mathcal{L}^{\Bumpeq}$ to denote the set of all sentences of such a richer language.

Definition 5 (Truth Conditions for $\mathcal{L}^{\Bumpeq}$) *Let a W-ordered model $\mathscr{W} = \langle W, R, V, \prec\rangle$ and let $w \in W$.*

- *$\mathcal{L}^{\Box}$-sentences are evaluated as usual (Definition 2);*
- *$\mathscr{W}, w \Vdash \Bumpeq_i\alpha$ if and only if for all w', if $w' \in \min_{\prec} R_i(w)$, then $\mathscr{W}, w' \Vdash \alpha$.*

The notions of satisfaction, truth (in a model), validity (in a class of models) and local entailment are generalised to $\mathcal{L}^{\Bumpeq}$-sentences and W-ordered models in the obvious way.

Informally, a sentence of the form $\Bumpeq_i\alpha$ holds in a world if α holds in all the most preferred amongst its i-accessible worlds. It is easy to see that $\Bumpeq$ is weaker than $\Box$, i.e., the following is a validity ($1 \leqslant i \leqslant n$):

$$\models \Box_i\alpha \rightarrow \Bumpeq_i\alpha$$

Hence, intuitively, flag can be read as *defeasible necessity*.

As an example, considering the W-ordered model $\mathscr{W}_1$ from Figure 5, we have that $\mathscr{W}_1, w_1 \Vdash \Bumpeq_a\neg p$ (but note that $\mathscr{W}_1, w_1 \not\Vdash \Box_a\neg p$).

Revisiting Preferential Modal Logics

In spite of its gain in expressiveness when checked against traditional approaches to defeasible reasoning, $\Bumpeq$ does not quite seem to allow us to formalise the type of reasoning motivated in the Introduction inasmuch as it relies on orderings on worlds. In this section, we shall revisit the framework for preferential modalities, in particular its semantic constructions.

R-Ordered Models

We start by giving a formal account of the semantic ideas put forward in the Introduction.

Definition 6 (*R*-Ordered Model) *An R-ordered model is a tuple $\mathscr{R} := \langle W, R, V, \ll\rangle$ where W is a (non-empty and possibly infinite) set of possible worlds, $R := \langle R_1, \ldots, R_n\rangle$, where each $R_i \subseteq W \times W$ is an accessibility relation on W, for $1 \leqslant i \leqslant n$, $V : W \longrightarrow \{0,1\}^{\mathcal{P}}$ is a valuation function assigning each world to a valuation on $\mathcal{P}$, and $\ll := \langle \ll_1, \ldots, \ll_n\rangle$, where each $\ll_i \subseteq R_i \times R_i$, for $1 \leqslant i \leqslant n$, is a well-founded strict partial order on the respective R_i, i.e., each $\ll_i$ is irreflexive, transitive and every non-empty $R_i' \subseteq R_i$ has minimal elements w.r.t. $\ll_i$ (see Definition 7).*

Given $\mathscr{R} := \langle W, R, V, \ll\rangle$, the intuition of W, R and V is the same as that in a standard Kripke model. The intuition of each $\ll_i$ in $\ll$ is that the pairs (w, w') that are lower down in the ordering $\ll_i$ are deemed as the most normal (or typical, or expected) in the context of R_i.

Definition 7 (Minimality w.r.t. $\ll_i$) *Let $\mathscr{R} = \langle W, R, V, \ll\rangle$ be an R-ordered model and let $X \subseteq R_i$, for some $1 \leqslant i \leqslant n$. Then $\min_{\ll_i} X := \{(w, w') \in X \mid$ there is no $(u, v) \in X$ such that $(u, v) \ll_i (w, w')\}$, i.e., $\min_{\ll_i} X$ denotes the minimal elements of X with respect to the preference relation $\ll_i$ associated to R_i.*

Since we assume each $\ll_i$ to be a well-founded strict partial order on the respective R_i, we are guaranteed that for every $X \subseteq R_i$ such that $X \neq \emptyset$, $\min_{\ll_i} X$ is well defined.

As an example, Figure 6 below depicts the R-ordered model $\mathscr{R}_1 := \langle W_1, R_1, V_1, \ll_1\rangle$, where $\langle W_1, R_1, V_1\rangle$ is as in Figure 4, and $\ll_1 := \langle \ll_a, \ll_b\rangle$, where $\ll_a := \{(w_1w_2, w_1w_3), (w_1w_3, w_4w_3), (w_1w_2, w_4w_3)\}$ and $\ll_b := \{(w_1w_4, w_2w_3)\}$, represented, respectively, by the dashed and the dotted arrows in the picture. (Note the direction of the $\ll$-arrows, which point from more preferred to less preferred transitions.) For the sake of readability, in our pictorial representations of R-ordered models, we shall omit the transitive $\ll$-arrows.

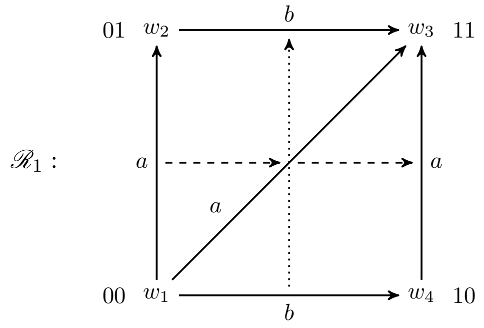


Figure 6: An R-ordered model for $\mathcal{P} = \{p, q\}$ and two modalities. The preference relation $\ll_a$ is represented by the dashed arrows, whereas $\ll_b$ by the dotted one.

A New Logic of Defeasible Modalities

We shall now enrich our underlying modal language with a family of additional modal operators $≋_i$, $1 \leqslant i \leqslant n$, where n is the number of classical modalities in the language. (For lack of a better term, we shall call $≋$ the 'banner'.) The sentences of the extended modal language are recursively defined as follows:

$$\alpha ::= p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \Box_i\alpha \mid\ ≋_i\alpha$$

With $\mathcal{L}^{≋}$ we shall denote the set of all sentences of the banner language.

Definition 8 *Let $\mathscr{R} = \langle W, R, V, \ll\rangle$. For every $w \in W$ and every $R_i \subseteq W \times W$, we define:*

$$R_i^w := \{(u, v) \mid (u, v) \in R_i \text{ and } u = w\}$$

Definition 9 ($\mathcal{L}^{≋}$ Truth Conditions) *Let $\mathscr{R} = \langle W, R, V, \ll\rangle$ be an R-ordered model and $w \in W$.*

- *$\mathcal{L}^{\Box}$-sentences are evaluated as usual;*
- *$\mathscr{R}, w \Vdash\ ≋_i\alpha$ if and only if for every w', if $(w, w') \in \min_{\ll_i} R_i^w$, then $\mathscr{R}, w' \Vdash \alpha$.*

The notions of satisfaction, truth (in a model), validity (in a class of models) and local entailment are also generalised to $\mathcal{L}^{\approxeq}$-sentences and *R*-ordered models in the usual way.

Informally, a sentence of the form $\approxeq_i\alpha$ holds in a world if α holds in all its most normally i-accessible worlds. As an example, in the *R*-ordered model $\mathscr{R}_1$ of Figure 6, we have that $\mathscr{R}_1, w_1 \Vdash \approxeq_a\neg p$ (but, of course, $\mathscr{R}_1, w_1 \nVdash \Box_a\neg p$).

Incidentally, $\approxeq$ too is weaker than $\Box$, as witnessed by the validity below ($1 \leqslant i \leqslant n$):

$$\models \Box_i\alpha \rightarrow \approxeq_i\alpha$$

Hence, $\approxeq$ provides an alternative perspective on the notion of defeasible necessity as formalised by $\widetilde{\Box}$. For instance, in an action context, some executions (which refer to transitions) of a given action are deemed as more normal than others. A priori, this is different from saying that some effects (which refer to target worlds) are normal. Indeed, an abnormal execution may still lead to the expected (normal) effect, just as a normal execution may produce an abnormal effect. (We shall come back to this issue later on.)

The definitions of *R*-ordered models and $\approxeq$, alongside the comment right above, raise the question as to how $\mathcal{L}^{\widetilde{\Box}}$ and $\mathcal{L}^{\approxeq}$ compare to each other in terms of expressive power. This is what we address in the next section.

Preferential Bisimulations

Standard bisimulations are used to determine whether two Kripke models have the same modal properties, and to reason about modal expressivity. Here, we extend the definition of bisimulations to *W*-ordered and *R*-ordered models, and use it to make precise the connection between these notions, and the resulting modalities and modal languages.

Definition 10 *Let $\mathscr{M} = \langle W, R, V\rangle$ and $\mathscr{M}' = \langle W', R', V'\rangle$. A bisimulation between $\mathscr{M}$ and $\mathscr{M}'$ is a non-empty binary relation E between their domains (that is, $\mathsf{E} \subseteq W \times W'$) such that, whenever $w\mathsf{E}w'$, we have that:*

1. *For every $p \in \mathcal{P}$, $\mathscr{M}, w \Vdash p$ if and only if $\mathscr{M}', w' \Vdash p$;*
2. *if wR_iv, then there exists a world v' in W' such that $v\mathsf{E}v'$ and $w'R'_iv'$, and*
3. *if $w'R'_iv'$, then there exists a world v in W such that $v\mathsf{E}v'$ and wR_iv.*

Informally, two worlds are bisimilar if they satisfy the same atomic information, and their modal accessibility structures match. Two pointed models $(\mathscr{M}, w)$ and $(\mathscr{M}', w')$ are bisimilar if there exists a bisimulation E between $\mathscr{M}$ and $\mathscr{M}'$ such that $w\mathsf{E}w'$. It then follows that:

Lemma 1 (Bisimulation invariance lemma) *If E is a bisimulation between $\mathscr{M} = \langle W, R, V\rangle$ and $\mathscr{M}' = \langle W', R', V'\rangle$, $w \in W$ and $w' \in W'$, and $w\mathsf{E}w'$, then w and w' satisfy the same basic modal sentences.*

The next definition and lemma generalise bisimulations to take account of a preference order on worlds, as defined on models of $\mathcal{L}^{\widetilde{\Box}}$. Informally, two worlds are bisimilar if they satisfy the same atomic information and their modal accessibility structures match, both with respect to accessible worlds and with respect to most preferred relative accessible worlds. Bisimilar worlds then also satisfy the same preferential modal sentences.

Definition 11 (*W*-ordered bisimulation) *Let W-ordered models $\mathscr{W} = \langle W, R, V, \prec\rangle$ and $\mathscr{W}' = \langle W', R', V', \prec'\rangle$. A bisimulation between $\mathscr{W}$ and $\mathscr{W}'$ is a non-empty binary relation $\mathsf{E} \subseteq W \times W'$ such that, whenever $w\mathsf{E}w'$, we have that:*

1. *For every $p \in \mathcal{P}$, $\mathscr{W}, w \Vdash p$ if and only if $\mathscr{W}', w' \Vdash p$;*
2. *if wR_iv, then there exists a world v' in W' such that $v\mathsf{E}v'$ and $w'R'_iv'$, and*
   - *if $v \in \min_{\prec} R_i(w)$, then $v' \in \min_{\prec'} R'_i(w')$;*
3. *if $w'R'_iv'$, then there exists a world v in W such that $v\mathsf{E}v'$ and wR_iv, and*
   - *if $v' \in \min_{\prec'} R'_i(w')$, then $v \in \min_{\prec} R_i(w)$.*

Lemma 2 (*W*-ordered bisimulation invariance lemma) *If E is a bisimulation between $\mathscr{W} = \langle W, R, V, \prec\rangle$ and $\mathscr{W}' = \langle W', R', V', \prec'\rangle$, and $w\mathsf{E}w'$, then w and w' satisfy the same modal sentences in the extended modal language $\mathcal{L}^{\widetilde{\Box}}$.*

Proof:
The lemma is proved by structural induction on $\alpha \in \mathcal{L}^{\widetilde{\Box}}$. We show that, for any $w \in W$ and $w' \in W'$, if $w\mathsf{E}w'$, then $\mathscr{W}, w \Vdash \alpha$ iff $\mathscr{W}', w' \Vdash \alpha$. For atomic propositions, and when $\alpha = \neg\beta$ or $\alpha = \beta_1 \vee \beta_2$, the proof is immediate. We consider the remaining two cases, namely when $\alpha = \Box_i\beta$ or $\alpha = \widetilde{\Box}_i\beta$.

Assume $\alpha = \Box_i\beta$ and let $\mathscr{W}, w \Vdash \Box_i\beta$. The proof is as for basic modal logic: Suppose $v' \in R'_i(w')$. Since $w\mathsf{E}w'$, there is some $v \in R_i(w)$ with $v\mathsf{E}v'$. Therefore $\mathscr{W}, v \Vdash \beta$, and hence $\mathscr{W}', v' \Vdash \beta$ by the induction hypothesis. It follows that $\mathscr{W}', w' \Vdash \Box_i\beta$. A symmetric argument applies if $\mathscr{W}', w' \Vdash \Box_i\beta$.

Assume $\alpha = \widetilde{\Box}_i\beta$ and let $\mathscr{W}, w \Vdash \widetilde{\Box}_i\beta$. Suppose $v' \in \min_{\prec} R'_i(w')$. Since $w\mathsf{E}w'$, there is some $v \in \min_{\prec} R_i(w)$ with $v\mathsf{E}v'$. Therefore $\mathscr{W}, v \Vdash \beta$, and hence $\mathscr{W}', v' \Vdash \beta$ by the induction hypothesis. It follows that $\mathscr{W}', w' \Vdash \widetilde{\Box}_i\beta$. A symmetric argument applies if $\mathscr{W}', w' \Vdash \Box_i\beta$. ■

We now turn to bisimulations between *R*-ordered models. As above, two worlds are bisimilar if they satisfy the same atomic information and their modal accessibility structures match, both in terms of accessible worlds and in terms of preference of accessibility.

Definition 12 (*R*-ordered bisimulation) *Let R-ordered models $\mathscr{R} = \langle W, R, V, \ll\rangle$ and $\mathscr{R}' = \langle W', R', V', \ll'\rangle$. A bisimulation between $\mathscr{R}$ and $\mathscr{R}'$ is a non-empty binary relation $\mathsf{E} \subseteq W \times W'$ such that, whenever $w\mathsf{E}w'$, we have that:*

1. *For every $p \in \mathcal{P}$, $\mathscr{R}, w \Vdash p$ if and only if $\mathscr{R}', w' \Vdash p$;*
2. *if wR_iv, then there exists a world v' in W' such that $v\mathsf{E}v'$ and $w'R'_iv'$, and*
   - *if $wv \in \min_{\ll_i} R_i^w$, then $w'v' \in \min_{\ll'_i} R_i'^{w'}$;*
3. *if $w'R'_iv'$, then there exists a world v in W such that $v\mathsf{E}v'$ and wR_iv, and*
   - *if $w'v' \in \min_{\ll'_i} R_i'^{w'}$, then $wv \in \min_{\ll_i} R_i^w$.*

Lemma 3 (*R*-ordered bisimulation invariance lemma) *If E is a bisimulation between $\mathscr{R} = \langle W, R, V, \lessdot \rangle$ and $\mathscr{R}' = \langle W', R', V', \lessdot' \rangle$, $w \in W$ and $w' \in W'$, and $w\mathsf{E}w'$, then w and w' satisfy the same modal sentences in the extended language $\mathcal{L}^{\approxeq}$.*

Proof:
The proof is by structural induction on $\alpha \in \mathcal{L}^{\approxeq}$ and is similar to that of Lemma 2. We show that, for any $w, w' \in W$, if $w\mathsf{E}w'$, then $\mathscr{R}, w \Vdash \alpha$ iff $\mathscr{R}', w' \Vdash \alpha$. We only prove the case when $\alpha = \approxeq_i \beta$.

Assume $\alpha = \approxeq_i \beta$ and let $\mathscr{R}, w \Vdash \approxeq_i \beta$. Suppose $w'v' \in \min_{\lessdot'_i} R_i'^{w'}$. Since $w\mathsf{E}w'$, there is some $wv \in \min_{\lessdot_i} R_i^w$ with $v\mathsf{E}v'$. Therefore $\mathscr{R}, v \Vdash \beta$, and hence $\mathscr{R}', v' \Vdash \beta$ by the induction hypothesis. It follows that $\mathscr{R}', w' \Vdash \approxeq_i \beta$. A symmetric argument applies if $\mathscr{R}', w' \Vdash \approxeq_i \beta$. ■

The relationship between $\mathcal{L}^{\approxeq}$ and $\mathcal{L}^{\backsimeq}$, and between *R*-ordered and *W*-ordered models, can be made precise using bisimulations. We first show that $\mathcal{L}^{\approxeq}$ is at least as expressive as $\mathcal{L}^{\backsimeq}$: Given a sentence $\alpha \in \mathcal{L}^{\backsimeq}$, let $\alpha^{\approxeq}$ be the sentence obtained by replacing all occurrences of $\backsimeq_i$ in α with $\approxeq_i$.

Definition 13 *Let $\mathscr{W} = \langle W, R, V, \prec \rangle$ be a W-ordered model. For any $u, v, w \in W$ such that wR_iu and wR_iv and $u \prec v$, let $wu \lessdot_i wv$. Then $\mathscr{R}_{\mathscr{W}} = \langle W, R, V, \lessdot \rangle$ is the R-ordered model induced by $\mathscr{W}$.*

Lemma 4 *For any $\alpha \in \mathcal{L}^{\backsimeq}$, $\mathscr{W} = \langle W, R, V, \prec \rangle$ and $w \in W$, $\mathscr{W}, w \Vdash \alpha$ if and only if in the R-ordered model $\mathscr{R}_{\mathscr{W}} = \langle W, R, V, \lessdot \rangle$ induced by $\mathscr{W}$, $\mathscr{R}_{\mathscr{W}}, w \Vdash \alpha^{\approxeq}$.*

Proof:
The proof is simple and proceeds by structural induction on the sentence α. ■

Lemma 4 shows that, if α and β are not equivalent in $\mathcal{L}^{\backsimeq}$, then their translations $\alpha^{\approxeq}$ and $\beta^{\approxeq}$ are also not equivalent in $\mathcal{L}^{\approxeq}$. Further, if $(\mathscr{W}, w)$ and $(\mathscr{W}', w')$ are distinguishable by some $\alpha \in \mathcal{L}^{\backsimeq}$, say, $\mathscr{W}, w \Vdash \alpha$ and $\mathscr{W}', w' \nVdash \alpha$, then $\mathscr{R}_{\mathscr{W}}$ and $\mathscr{R}'_{\mathscr{W}}$ are distinguishable by $\alpha^{\approxeq} \in \mathcal{L}^{\approxeq}$. Hence, $\mathcal{L}^{\approxeq}$ is at least as expressive as $\mathcal{L}^{\backsimeq}$.

The converse of this result may not be as obvious to see, and translating *R*-ordered models to *W*-ordered models requires more care. The light switch example (Figure 1) shows that, even in the case of a single modality, there is no direct translation of a preference order on *R* to a preference order on *W*. There is no order on the two worlds w_1 and w_2 such that w_1 is the preferred result of toggling the light switch when the light is off, but w_2 is the preferred result when the light is on. A further problematic aspect is that *R*-ordered models allow for a preference order on each accessibility relation, whereas a *W*-ordered semantics assume a single common preference order on worlds.

Definition 14 *Let $\mathscr{R} = \langle W, R, V, \lessdot \rangle$ be an R-ordered model with single accessibility relation R_1. Let $W' = W \times W$; let $V'(uw) = V(w)$; let uvR'_1vw whenever vR_1w, and let $uv \prec u'v'$ whenever $uv \lessdot u'v'$. Then $\mathscr{W}_{\mathscr{R}} = \langle W', R', V', \prec \rangle$ is the W-ordered model induced by $\mathscr{R}$.*

As an example, we apply Definition 14 to obtain the *W*-ordered models induced by the models of Figures 1 and 2, and depicted in Figure 7 and Figure 8 respectively. Note that in Figure 7, $w_1w_2 \prec w_1w_1$ and $w_2w_1 \prec w_2w_2$, reflecting the intuition of normal execution of the action as an order on worlds. In Figure 8, the order on worlds is reversed, with $w_1w_1 \prec w_1w_2$ and $w_2w_2 \prec w_2w_1$, depicting the intuition of defeasible knowledge of the agent as an order on worlds.

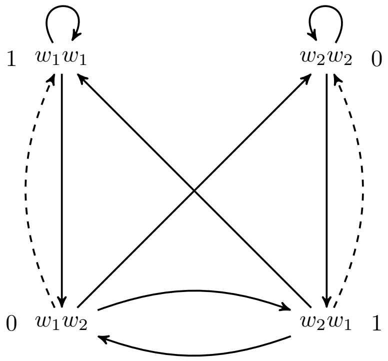


Figure 7: The induced *W*-ordered possible-worlds model for one action (toggle) and one atom (on).

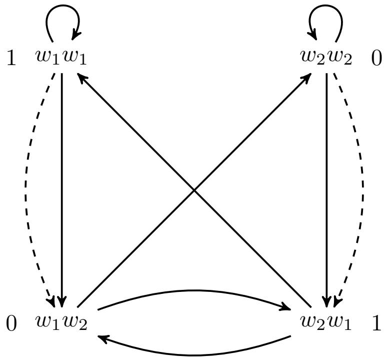


Figure 8: The induced *W*-ordered possible-worlds model for one agent (M) and one atom (correct).

Theorem 1 *Let $\mathscr{R} = \langle W, R, V, \lessdot \rangle$ be an R-ordered model with a single accessibility relation R_1 and let $\mathscr{W}_{\mathscr{R}} = \langle W, R, V, \prec \rangle$ be the W-ordered model induced by $\mathscr{R}$. Let $\mathscr{R}_{\mathscr{W}_{\mathscr{R}}} = \langle W', R', V', \lessdot' \rangle$ be the R-ordered model induced by $\mathscr{W}_{\mathscr{R}}$. Then there is a full bisimulation between $\mathscr{R}$ and $\mathscr{R}_{\mathscr{W}_{\mathscr{R}}}$, i.e., with domain W and range $W \times W$.*

Proof:
Let E be defined by: $w\mathsf{E}vw$ for all $v, w \in W$. We need to show that E is a full bisimulation relation. So, let $u, v \in W$. Then $v\mathsf{E}uv$.

1. It follows immediately from the construction of $\mathscr{R}_{\mathscr{W}_{\mathscr{R}}}$ that v and uv satisfy the same atomic propositions.
2. Suppose vR_1w. It follows again from the construction of $\mathscr{R}_{\mathscr{W}_{\mathscr{R}}}$ that uvR'_1vw and $w\mathsf{E}vw$. Further, if $vw \in \min_{\lessdot_1} R_1^v$, then $vw \in \min_{\prec} R_1(uv)$, and hence $vw \in \min_{\lessdot'_1} (R'_1)^{uv}$.

3. Suppose uvR'_1vw. It again follows from the construction of $\mathscr{R}_{\mathscr{W}_{\mathscr{R}}}$ that vR_1w and $w\mathsf{E}vw$. Further, if $vw \in \min_{\lessdot'_1}(R'_1)^{uv}$, then $vw \in \min_{\prec} R_1(w)$, and hence $vw \in \min_{\lessdot_1} R_1^v$.

■

We illustrate the construction of Theorem 1 by applying Definition 13 to the induced *W*-ordered model in Figure 7 to obtain the *R*-ordered model of Figure 9. In Figure 9, the dashed arrows represent the preference order $\lessdot'$. Theorem 1 then states that the *R*-ordered model of Figure 1 (with the order as described in the Introduction) is bisimilar to the *R*-ordered model of Figure 9, The construction is via the *W*-ordered model of Figure 7.

Similarly, the *R*-ordered model of Figure 2 (again, with the order as described in the Introduction) is bisimilar to the *R*-ordered model of Figure 10, which is constructed via the *W*-ordered model of Figure 8.

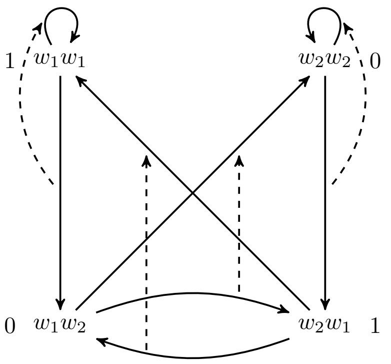


Figure 9: The induced bisimilar *R*-ordered model for one action (toggle) and one atom (on).

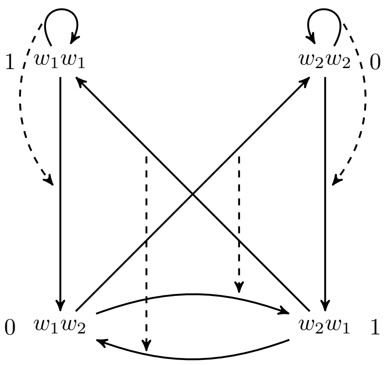


Figure 10: The induced bisimilar *R*-ordered model for one agent (M) and one atom (correct).

Corollary 1 *$\mathcal{L}^{\approxeq}$ and $\mathcal{L}^{\backsimeq}$ can distinguish between the same modal propositions when restricted to a single modality.*

Proof:
The bisimulation result of Theorem 1 shows that any *R*-ordered model is bisimilar to some *R*-ordered model induced by a *W*-ordered model. Lemma 3 ensures that bisimilar worlds satisfy the same modal sentences, and that bisimilar models can distinguish between the same modal properties. We need therefore consider only *R*-ordered models induced by some *W*-ordered model when reasoning about expressivity. The result then follows from Lemma 4. ■

Corollary 1 may be seen as a negative result in the sense that, at least in the monomodal case, no richer language is obtained when substituting a preference order on the accessibility relation for the preference order on worlds. It is also clear that the results of Theorem 1 and Corollary 1 can be generalised to multi-modal languages if multiple preference relations on *W* are allowed.

What, then, has been gained? As we have argued, there are a number of contexts in which an order on the accessibility relation has an intuitive appeal. The induced *W*-ordered models of Definition 13 are technically useful, but intuitively hard to motivate. However, from an implementation perspective, we now know that a reasoner based on a *W*-ordered semantics suffices also for reasoning over *R*-ordered models. This, together with our previous results (Britz and Varzinczak 2013), establish the following:

Corollary 2 *Satisfiability checking for monomodal $\mathcal{L}^{\approxeq}$ is* PSPACE*-complete.*

Discussion and Related Work

It might be worth emphasising that the logics we have investigated here do not aim at providing a formal account of the notion of *most*, as addressed in the study of generalised quantifiers (Lindström 1966) and, more recently, in a modal context by Veloso *et al.* (2009) and Askounis *et al.* (2012). Clearly, they are not about degrees of truth, as it has been studied in fuzzy logics, nor about degrees of possibility and necessity, as addressed by possibilistic logics (Dubois, Lang, and Prade 1994). Instead, here we have investigated a rather complementary notion to those ones, namely that of *normal*, *expected*, *practical* necessity, which need not rely on majority or degrees of likelihood.

In a sense, the notions we investigated here can be seen as the qualitative counterpart of possibilistic modalities (Liau 1999; Liau and Lin 1996). (We thank an anonymous referee for pointing this out to us.) There, each possible world w is associated with a *possibility distribution* $\pi_w : W \longrightarrow [0, 1]$, the intuition of which is to capture the degree of likelihood (in terms of belief) of all possible worlds w.r.t. w. In that setting, the pairs (w, w') for which $\pi_w(w')$ is maximal correspond here to the most preferred pairs in a single accessibility relation. In this sense, there are strong links between *monomodal* $\approxeq$ and the preferential possibilistic semantics for epistemic reasoning.

Currently, the definition of *R*-ordered model (Definition 6) allows only for elements of the same accessibility relation R_i to be ordered (via the respective $\lessdot_i$). More generally, we could have defined $\lessdot$ as a relation on $\bigcup_{1\leqslant i\leqslant n} R_i \times$

$\bigcup_{1 \leqslant i \leqslant n} R_i$, so that we allow pairs (w, w') belonging to different *R*-components to be compared as well. An investigation of the philosophical and practical ramifications of this alternative definition is left for future work.

We have seen that one can obtain *R*-ordered models from *W*-ordered models by inducing an ordering on edges from the ordering on worlds. The result is an 'embedding' of $\tilde{\sqcup}$ into $\mathcal{L}^{≋}$. Conversely, in the monomodal case, we can obtain *W*-ordered models from *R*-ordered models by inducing an ordering on worlds from an ordering on edges. If we allow multiple preferences on worlds, the latter result can easily be generalised, thereby establishing that $\mathcal{L}^{≋}$ and $\mathcal{L}^{\tilde{\sqcup}}$ are equally expressive. This would have an interesting consequence, namely that the notions of 'normal effects' and 'normal executions' of actions are one and the same. This a priori counter-intuitive claim is easily justifiable. It turns out the effects of an action (the worlds one 'lands' in) depend to a large extent on what the current state of the world (the 'departing' points) is. In other terms, talking about effects (tacitly) amounts to talking about pairs (w, w'), linking both a context of execution and the action's outcome. This feature just carries over when normality is considered.

In this work, we have not addressed the question as to what an appropriate notion of entailment for $\mathcal{L}^{≋}$ is and have contented ourselves with the standard (Tarskian) definition, which is monotonic (and therefore inappropriate in many contexts). The recent results by Booth *et al.* (2015) in a propositional setting may provide us with a springboard to investigate this matter in more expressive languages such as those we are interested in here.

Outlook on Further Work

We shall now briefly discuss about possible ideas for exploration stemming from the present work.

R-based Conditionals

A framework for representing and reasoning with defeasibility would not be complete without an account of (defeasible) conditionals. Here we catch a glimpse of two versions thereof which can both be defined in our *R*-ordered models semantics.

Given an *R*-ordered model $\mathscr{R}$, for every propositional sentence α, let $R_\alpha := \{(w, w') \mid \mathscr{R}, w \Vdash \alpha \text{ and } \mathscr{R}, w' \Vdash \alpha\}$ and $\ll_\alpha$ its corresponding preference relation. (Of course, if we work in a finite propositional language, then there are finitely many of such R_αs and $\ll_\alpha$s.) We can then define a conditional statement as a macro in $\mathcal{L}^{≋}$ as follows:

- $\alpha \leadsto_1 \beta$ if and only if $≋_\alpha \beta$.

Such a definition, of course, has its limitations, as it only allows for propositional sentences in the antecedent of the conditional. A generalisation to the case where $\alpha \in \mathcal{L}^{≋}$ would hardly improve matters, as we would end up with an infinite number of accessibility relations in the *R*-component of our *R*-ordered models.

Fortunately, we can do better than this. First, we need to define an extra, identity relation *id* on *W* and order its elements in the same way as for the other *R*-components. The intuition of doing so is that the most normal *id*-arrows correspond to the most normal worlds, i.e., we get an ordering on worlds induced by the ordering on the elements of the identity relation. With this, we can define our second candidate for a conditional in the following way. First, for every $\alpha \in \mathcal{L}^{≋}$, let $id^\alpha := \{(w, w) \in id \mid \mathscr{R}, w \Vdash \alpha\}$. Then

- $\mathscr{R} \Vdash \alpha \leadsto_2 \beta$ if and only if for every w such that $(w, w) \in \min_{\ll_{id}} id^\alpha$, it holds that $\mathscr{R}, w \Vdash \beta$.

We shall leave an investigation of the appropriateness of $\leadsto_2$ as a defeasible conditional for future work.

Next Steps in Preferential Reasoning for DLs

In the context of formal ontologies specified in Description Logics (Baader et al. 2007), placing a preference order on binary relations as we proposed here has a natural appeal. As an example, consider the role name hasChild: 'Normal' tuples in this relation may be biological or adopted parent-child tuples, while an 'exceptional' tuple may be an appointed legal guardian parent-child tuple. In this example, there is nothing exceptional about either the legal guardian or the child—the exceptionality lies in the nature of their *relationship*.

To make things more precise, given a DL interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, we can enrich it with a collection of preference relations $\ll^{\mathcal{I}} := \langle \ll^{\mathcal{I}}_{r_1}, \ldots, \ll^{\mathcal{I}}_{r_n} \rangle$, one for each role name and each of which satisfying the conditions in Definition 6. Armed with this semantic construction, it becomes possible to:

- Define *defeasible value restrictions* (Britz et al. 2013) of the form $\forall r.C$, like $\forall$hasChild.Male, which refers to those individuals whose most normal parenting relations are of male children;
- State *defeasible role inclusions* of the form $r_1 \mathrel{\underset{\sim}{\sqsubset}} r_2$, as in e.g. parentOf $\mathrel{\underset{\sim}{\sqsubset}}$ progenitorOf, which stipulates that the role of being a parent is usually that of also being the progenitor;
- State *typicality-based role instances* in the ABox of the form $\bullet r(a, b)$, where $\bullet$ is the extension of a typicality operator (Booth, Meyer, and Varzinczak 2012; Giordano et al. 2007) to roles, like •hasChild(john, anne), conveying the information that, under the interpretation of role hasChild, the tuple (john, anne) is to be regarded as a typical one;
- State *defeasible role properties* like in saying that role marriedTo is *normally functional* and that partOf is *normally transitive*, while allowing for exceptions, i.e., less normal tuples failing the relation's property under consideration.

Moreover, definitions analogous to those in the preceding subsection would allow us to:

- State *defeasible concept subsumptions* (Britz, Heidema, and Meyer 2008; Britz, Meyer, and Varzinczak 2011b; Casini and Straccia 2010; Giordano et al. 2007) of the form $C \mathrel{\underset{\sim}{\sqsubset}} D$, as in Mother $\mathrel{\underset{\sim}{\sqsubset}}$ $\exists$marriedTo, of which the intuition is that usually, mothers are married.

It is an open question whether a result similar to that obtained in Theorem 1 holds in a DL context. Roles can be reified, similar to the reification of n-ary relations in DLs (Sattler, Calvanese, and Molitor 2007), as a workaround to model preferences on tuples as preferences on objects in a DL enriched with a preferential subsumption relation $\underset{\sim}{\sqsubset}$. Nevertheless, it is not immediately clear how the addition of preferential roles to a DL with preferential subsumption would affect its expressivity.

Defeasible Comparative Epistemic Logic

By placing a preference relation on the accessibility relations, we can get to a generalisation of Comparative Epistemic Logic (CEL) (Ditmarsch, Hoek, and Kooi 2012).

In CEL, a statement of the form $a \geq b$ intuitively means "agent b knows at least as much as agent a". The corresponding semantics is given by:

- $\mathscr{M}, w \Vdash a \geq b$ if and only if $R_b(w) \subseteq R_a(w)$.

In the context of our enriched semantic framework, we could envisage making statements of the form "agent b *normally* knows as much as agent a", of which a semantics can be given by the condition $\min_{\ll_b} R_b^w \subseteq R_a^w$.

Summary and Conclusion

The contributions of the present paper can be summarised as follows: (*i*) the motivation for and the definition of a semantic structure allowing for the ordering of *pairs* of worlds (instead of worlds *tout court*, as is customary in traditional NMR formalisms) and (*ii*) a generalisation of bisimulation to the preferential case together with a result relating our new semantics to that we studied in previous work and showing that, in the *monomodal* case, they are equivalent.

We have introduced a logic allowing for modal operators the intuition of which is to capture the idea of some transitions being more normal than others. As we have seen, our *R*-ordered models can be used to provide the extended language with an intuitive and elegant semantics. The resulting framework provides for an alternative formalisation for the notion of defeasible necessity we studied previously.

We have given examples, in an action, epistemic and deontic contexts, of what this semantic structure, as simple as it is, would allow us to represent (or give a meaning to) that one cannot do with standard Kripkean semantics. Likewise, we have briefly illustrated the fruitfulness of our definitions in other formalisms, in particular in a DL setting.

Acknowledgements

This work is based upon research supported in part by the Brazilian National Council for Scientific and Technological Development (CNPq) under grant number 302002/2014-6. This work was also partially funded by the National Research Foundation of South Africa (UIDs 81225 and 85482, IFR1202160021 and IFR2011032700018).

References

Askounis, D.; Koutras, C.; and Zikos, Y. 2012. Knowledge means 'all', belief means 'most'. In Fariñas del Cerro, L.; Herzig, A.; and Mengin, J., eds., *Proceedings of the 13th European Conference on Logics in Artificial Intelligence (JELIA)*, number 7519 in LNCS, 41–53. Springer.

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P., eds. 2007. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2 edition.

Baltag, A., and Smets, S. 2006. Dynamic belief revision over multi-agent plausibility models. In van der Hoek, W., and Wooldridge, M., eds., *Proceedings of LOFT*, 11–24. University of Liverpool.

Baltag, A., and Smets, S. 2008. A qualitative theory of dynamic interactive belief revision. In Bonanno, G.; van der Hoek, W.; and Wooldridge, M., eds., *Logic and the Foundations of Game and Decision Theory (LOFT7)*, number 3 in Texts in Logic and Games, 13–60. Amsterdam University Press.

Blackburn, P.; Benthem, J.; and Wolter, F. 2006. *Handbook of Modal Logic*. Elsevier North-Holland.

Booth, R.; Casini, G.; Meyer, T.; and Varzinczak, I. 2015. On the entailment problem for a logic of typicality. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*.

Booth, R.; Meyer, T.; and Varzinczak, I. 2012. PTL: A propositional typicality logic. In Fariñas del Cerro, L.; Herzig, A.; and Mengin, J., eds., *Proceedings of the 13th European Conference on Logics in Artificial Intelligence (JELIA)*, number 7519 in LNCS, 107–119. Springer.

Boutilier, C. 1994. Conditional logics of normality: A modal approach. *Artificial Intelligence* 68(1):87–154.

Britz, K., and Varzinczak, I. 2013. Defeasible modalities. In *Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge (TARK)*, 49–60.

Britz, K.; Casini, G.; Meyer, T.; and Varzinczak, I. 2013. Preferential role restrictions. In *Proceedings of the 26th International Workshop on Description Logics*, 93–106.

Britz, K.; Heidema, J.; and Meyer, T. 2008. Semantic preferential subsumption. In Lang, J., and Brewka, G., eds., *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 476–484. AAAI Press/MIT Press.

Britz, K.; Meyer, T.; and Varzinczak, I. 2011a. Preferential reasoning for modal logics. *Electronic Notes in Theoretical Computer Science* 278:55–69. Proceedings of the 7th Workshop on Methods for Modalities (M4M'2011).

Britz, K.; Meyer, T.; and Varzinczak, I. 2011b. Semantic foundation for preferential description logics. In Wang, D., and Reynolds, M., eds., *Proceedings of the 24th Australasian Joint Conference on Artificial Intelligence*, number 7106 in LNAI, 491–500. Springer.

Casini, G., and Straccia, U. 2010. Rational closure for defeasible description logics. In Janhunen, T., and Niemelä, I., eds., *Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA)*, number 6341 in LNCS, 77–90. Springer-Verlag.

Ditmarsch, H.; Hoek, W.; and Kooi, B. 2012. Local properties in modal logic. *Artificial Intelligence* 187:133–155.

Dubois, D.; Lang, J.; and Prade, H. 1994. Possibilistic logic. In Gabbay, D.; Hogger, C.; and Robinson, J., eds., *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3. Oxford University Press. 439–513.

Giordano, L.; Olivetti, N.; Gliozzi, V.; and Pozzato, G. 2007. Preferential description logics. In Dershowitz, N., and Voronkov, A., eds., *Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, number 4790 in LNAI, 257–272. Springer.

Hansson, B. 1969. An analysis of some deontic logics. *Noûs* 3:373–398.

Katsuno, H., and Mendelzon, A. 1991. Propositional knowledge base revision and minimal change. *Artificial Intelligence* 3(52):263–294.

Kraus, S.; Lehmann, D.; and Magidor, M. 1990. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence* 44:167–207.

Lehmann, D., and Magidor, M. 1992. What does a conditional knowledge base entail? *Artificial Intelligence* 55:1–60.

Lewis, D. 1973. *Counterfactuals*. Blackwell.

Lewis, D. 1974. Semantic analyses for dyadic deontic logic. In Stenlund, S., ed., *Logical Theory and Semantic Analysis*. D. Reidel Publishing Company. 1–14.

Liau, C.-J., and Lin, B.-P. 1996. Possibilistic reasoning–a mini-survey and uniform semantics. *Artificial Intelligence* 88(1-2):163–193.

Liau, C.-J. 1999. On the possibility theory-based semantics for logics of preference. *International Journal of Approximate Reasoning* 20(2):173–190.

Lindström, P. 1966. First-order predicate logic with generalized quantifiers. *Theoria* 32:286–195.

Makinson, D. 2005. *Bridges from Classical to Nonmonotonic Logic*, volume 5 of *Texts in Computing*. King's College Publications.

Sattler, U.; Calvanese, D.; and Molitor, R. 2007. Relationships with other formalisms. In Baader et al. (2007). chapter 4, 149–192.

Shoham, Y. 1988. *Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence*. MIT Press.

Stalnaker, R. 1968. A theory of conditionals. In Rescher, N., ed., *Studies in Logical Theory*. Blackwell. 98–112.

Veloso, P.; Veloso, S.; Viana, J.; de Freitas, R.; Benevides, M.; and Delgado, C. 2009. On vague notions and modalities: a modular approach. *Logic Journal of the IGPL* 18(3):381–402.

Properties of ABA$^+$ for Non-Monotonic Reasoning

Kristijonas Čyras and **Francesca Toni**
Imperial College London, UK

Abstract

We investigate properties of ABA$^+$, a formalism that extends the well studied structured argumentation formalism Assumption-Based Argumentation (ABA) with a preference handling mechanism. In particular, we establish desirable properties that ABA$^+$ semantics exhibit. These pave way to the satisfaction by ABA$^+$ of some (arguably) desirable principles of preference handling in argumentation and non-monotonic reasoning, as well as non-monotonic inference properties of ABA$^+$ under various semantics.

1 Introduction

Recent decades have seen a number of non-monotonic reasoning (NMR) formalisms advanced (see e.g. (Brewka, Niemelä, and Truszczyński 2007) for an overview). Since preferences are ubiquitous in common-sense reasoning, there has been a considerable effort to integrate preference information within NMR formalisms (cf. e.g. (Brewka, Truszczyński, and Niemelä 2008; Delgrande et al. 2004; Domshlak et al. 2011; Kaci 2011)). To evaluate distinct formalisms, various properties of both non-monotonic inference and preference handling have been proposed, see e.g. (Makinson 1988; Kraus, Lehmann, and Magidor 1990; Brewka and Eiter 1999; Brewka, Truszczyński, and Woltran 2010; Šimko 2014).

Meanwhile, argumentation (as overviewed in (Rahwan and Simari 2009)) has become an established branch of AI widely used for NMR (see e.g. (Dung 1995; Bondarenko et al. 1997; Modgil and Prakken 2013)). Broadly speaking, information in argumentation is represented via *arguments*, while *attacks* among them indicate conflicts. Procedures, known as *argumentation semantics*, are employed to select *extensions*, i.e. sets of collectively acceptable arguments. Preferences in argumentation also play a significant role (cf. e.g. (Simari and Loui 1992; Kaci 2011)), by allowing to, for instance, discriminate among arguments or extensions. Over the years, numerous formalisms of argumentation with preferences have been presented (see Section 7) and some properties for argumentation with preferences indicated (e.g. (Brewka, Truszczyński, and Woltran 2010; Modgil and Prakken 2013; Amgoud and Vesic 2014; Dung 2016)).

NMR properties are also adaptable to argumentation setting. For example, the well known non-monotonic inference properties of *Cautious Monotonicity* and *Cumulative Transitivity* (cf. (Makinson 1988; Kraus, Lehmann, and Magidor 1990)) concern what happens when a conclusion reached through a reasoning process is added to the knowledge base to reason with anew. These properties have been cast with respect to extensions in argumentation, in e.g. (Čyras and Toni 2015; Dung 2016).

Preference handling properties for NMR can be phrased in terms of extensions in argumentation too. For instance, the well known Principle I from (Brewka and Eiter 1999) regarding preferred answer sets can be applied to argumentation semantics thus: if two extensions E_1 and E_2 coincide except for two arguments $\mathsf{A} \in E_1 \setminus E_2$ and $\mathsf{B} \in E_2 \setminus E_1$ such that A is preferred over B, then E_2 should not be chosen as a 'preferable' extension. Likewise, a common property of NMR says that, in the absence of preference information, a formalism extended with a preference handling mechanism should return the same extensions as the preference-free version of the formalism (see e.g. (Brewka, Truszczyński, and Woltran 2010; Šimko 2014)).

In this paper, drawing from the above mentioned works, we investigate various properties of a recently proposed NMR formalism ABA$^+$ (Čyras and Toni 2016). ABA$^+$ extends with a preference handling mechanism a well established argumentation formalism, Assumption-Based Argumentation (ABA) (Bondarenko et al. 1997; Toni 2014). Whereas a common way to approach preferences in argumentation is to use preference information to *discard* the attacks from arguments that are less preferred than the ones they attack (see e.g. (Amgoud and Cayrol 2002; Bench-Capon 2003; Kaci and van der Torre 2008; Brewka et al. 2013; Besnard et al. 2014)), ABA$^+$ instead *reverses* such attacks. We show that ABA$^+$'s method of accounting for preferences satisfies (arguably) desirable properties.

On the one hand, we consider preference handling properties from (Brewka and Eiter 1999; Brewka, Truszczyński, and Woltran 2010; Amgoud and Vesic 2014) and show their satisfaction under various ABA$^+$ semantics. On the other hand, building on the investigations of Cumulative Transitivity and Cautious Monotonicity for ABA (Čyras and Toni 2015), we analyse ABA$^+$ in the light of these non-monotonic inference properties, and show that results obtained for ABA carry over to ABA$^+$. In addition, we make use of the well known principle of *Contraposition* of rules (see e.g. (Modgil and Prakken 2013)) and prove it guarantees that ABA$^+$ semantics satisfy desirable properties akin to those in e.g. (Dung 1995; Bondarenko et al. 1997; Modgil and Prakken 2013).

The paper is organized as follows. Sections 2 and 3 give preliminaries on ABA and ABA$^+$. In Section 4 ABA$^+$ semantics are analysed. Preference handling properties of ABA$^+$ are studied in Section 5, while Section 6 concerns ABA$^+$ and non-monotonic inference properties. After discussing related work (Section 7), we conclude in Section 8.

2 Preliminaries

We base the following ABA background on (Toni 2014).

Definition 1. An *ABA framework* is a tuple $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$, where:

- $(\mathcal{L}, \mathcal{R})$ is a deductive system with a language $\mathcal{L}$ and a set $\mathcal{R}$ of rules of the form $\varphi_0 \leftarrow \varphi_1, \ldots, \varphi_m$ with $m \geqslant 0$ and $\varphi_i \in \mathcal{L}$ for $i \in \{0, \ldots, m\}$; φ_0 is referred to as the *head* of the rule, and $\varphi_1, \ldots, \varphi_m$ is referred to as the *body* of the rule; if $m = 0$, then the rule $\varphi_0 \leftarrow \varphi_1, \ldots, \varphi_m$ is written as $\varphi_0 \leftarrow \top$ and is said to have an empty body;
- $\mathcal{A} \subseteq \mathcal{L}$ is a non-empty set, whose elements are referred to as *assumptions*;
- $\overline{} : \mathcal{A} \to \mathcal{L}$ is a total map: for $\alpha \in \mathcal{A}$, the $\mathcal{L}$-formula $\overline{\alpha}$ is referred to as the *contrary* of α.

We focus on *flat* ABA frameworks, where no assumption is the head of any rule. Flat ABA frameworks are very common, and capture, as instances, widely used paradigms of non-monotonic reasoning, such as Logic Programming and Default Logic (see e.g. (Bondarenko et al. 1997)).

Definition 2. A *deduction for $\varphi \in \mathcal{L}$ supported by $S \subseteq \mathcal{L}$ and $R \subseteq \mathcal{R}$*, denoted by $S \vdash^R \varphi$, is a finite tree with the root labelled by φ, leaves labelled by $\top$ or elements from S, the children of non-leaf nodes ψ labelled by the elements of the body of some rule from $\mathcal{R}$ with head ψ, and R being the set of all such rules. For $E \subseteq \mathcal{L}$, the *conclusions* $Cn(E)$ of E is the set of elements with deductions supported by $S \subseteq E$ and some $R \subseteq \mathcal{R}$, i.e. $Cn(E) = \{\varphi \in \mathcal{L} : \exists\, S \vdash^R \varphi,\ S \subseteq E,\ R \subseteq \mathcal{R}\}$.

Assumption-level attacks in ABA are defined thus.

Definition 3. A set $A \subseteq \mathcal{A}$ *attacks* a set $B \subseteq \mathcal{A}$, denoted $A \leadsto B$, if there is a deduction $A' \vdash^R \overline{\beta}$, for some $\beta \in B$, supported by some $A' \subseteq A$ and $R \subseteq \mathcal{R}$. For $E \subseteq \mathcal{A}$, also called an *extension*, we say that:

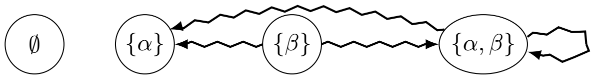
- E is *conflict-free* if $E \not\leadsto E$;
- E *defends* $\alpha \in \mathcal{A}$ if for all $B \leadsto \{\alpha\}$ it holds that $E \leadsto B$;
- E is *admissible* if E is conflict-free and defends all $\alpha \in E$.

The most standard ABA semantics are as follows.

Definition 4. A conflict-free set $E \subseteq \mathcal{A}$ is:

- *stable*, if $E \leadsto \{\beta\}$ for every $\{\beta\} \subseteq \mathcal{A} \setminus E$;
- *complete* if E is admissible and contains every assumption it defends;
- *preferred* if E is $\subseteq$-maximally admissible;
- *grounded* if E is $\subseteq$-minimally complete;
- *ideal* if E is $\subseteq$-maximal such that E is admissible and contained in all preferred extensions.

Example 5. Let $\mathcal{L} = \{\alpha, \beta, \overline{\alpha}, \overline{\beta}\}$, $\mathcal{R} = \{\overline{\alpha} \leftarrow \beta\}$ and $\mathcal{A} = \{\alpha, \beta\}$. In $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$, $\{\beta\}$ attacks both $\{\alpha\}$ and $\{\alpha, \beta\}$, while $\{\alpha, \beta\}$ attacks itself and $\{\alpha\}$. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ can be graphically represented via its *assumption framework*, pictured below (in illustrations of assumption frameworks, nodes hold sets of assumptions while directed edges indicate attacks):



This $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ has a unique complete extension $\{\beta\}$, which is also grounded, ideal, preferred and stable, and has conclusions $Cn(\{\beta\}) = \{\overline{\alpha}, \beta\}$.

3 ABA$^+$

ABA$^+$ (Čyras and Toni 2016) extends ABA with preferences as follows.

Definition 6. An *ABA$^+$ framework* is any tuple $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, where $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ is an ABA framework and $\leqslant$ is a preorder (i.e. a transitive and reflexive binary relation) on $\mathcal{A}$.

Differently from e.g. (Modgil and Prakken 2013; 2014; García and Simari 2014), ABA$^+$ considers preferences on assumptions rather than (defeasible) rules. This is not, however, a conceptual difference, since assumptions are the only defeasible component in ABA$^+$.

Unless stated differently, we consider a fixed, but otherwise arbitrary ABA$^+$ framework $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, and implicitly assume $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ to be its underlying ABA framework. The strict counterpart $<$ of $\leqslant$ is defined as $\alpha < \beta$ iff $\alpha \leqslant \beta$ and $\beta \nleqslant \alpha$, for any α and β.

ABA$^+$ attack relation is given thus.

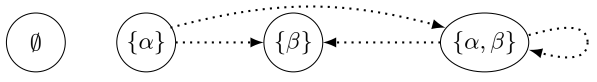
Definition 7. A set $A \subseteq \mathcal{A}$ of assumptions *$<$-attacks* a set $B \subseteq \mathcal{A}$ of assumptions, written as $A \leadsto_< B$, if:

- either there is a deduction $A' \vdash^R \overline{\beta}$, for some $\beta \in B$, supported by $A' \subseteq A$, and $\nexists \alpha' \in A'$ with $\alpha' < \beta$;
- or there is a deduction $B' \vdash^R \overline{\alpha}$, for some $\alpha \in A$, supported by $B' \subseteq B$, and $\exists \beta' \in B'$ with $\beta' < \alpha$.

The first type of attack is called *normal*, and the second one *reverse*.

ABA$^+$ requires a standard ABA attack to be reversed whenever the attacker has an assumption less preferred than the one attacked. The following example illustrates.

Example 8. Recall $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ from Example 5. Suppose $\beta < \alpha$. In the ABA$^+$ framework $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, $\{\beta\}$ 'tries' to attack $\{\alpha\}$, but is prevented by the preference $\beta < \alpha$. Instead, $\{\alpha\}$ $<$-attacks $\{\beta\}$, and likewise $\{\alpha, \beta\}$, via reverse attack, and the latter $<$-attacks both itself and $\{\beta\}$ via reverse attack. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ can be represented graphically as follows (reverse attacks in assumption frameworks will be denoted by dotted arrows):



In contrast with the ABA framework, where $\{\beta\}$ is unattacked and generates an attack on $\{\alpha\}$, in the

ABA$^+$ framework, $\{\alpha\}$ is $<$-unattacked and $<$-attacks all sets of assumptions that contain β. This concords with the intended meaning of the preference $\beta < \alpha$, that the conflict should be resolved in favour of α.

This concept of $<$-attack reflects the interplay between deductions, contraries and preferences, by representing inherent conflicts among sets of assumptions while accounting for preference information. Normal attacks follow the standard notion of attack in ABA, additionally, preventing the attack to succeed when the attacker uses assumptions less preferred than the one attacked. Reverse attacks, meanwhile, resolve the conflict between two sets of assumptions by favouring the one containing an assumption whose contrary is deduced, over the one which uses less preferred assumptions to deduce that contrary.

The notions of conflict-freeness and defence w.r.t. $\rightsquigarrow_<$, and ABA$^+$ semantics are given as follows.

Definition 9. For $E \subseteq \mathcal{A}$ we say that:

- E is *$<$-conflict-free* if $E \not\rightsquigarrow_< E$;
- E *$<$-defends* $\alpha \in \mathcal{A}$ if for all $B \rightsquigarrow_< \{\alpha\}$ it holds that $E \rightsquigarrow_< B$; and
- E is *$<$-admissible* if E is $<$-conflict-free and $<$-defends every $\alpha \in E$.

In Example 8, $\emptyset$, $\{\alpha\}$ and $\{\beta\}$ are conflict-free in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ and $<$-conflict-free in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, whereas $\{\alpha, \beta\}$ is not ($<$-)conflict-free in either framework.

Definition 10. A $<$-conflict-free extension $E \subseteq \mathcal{A}$ is:

- *$<$-stable* if $E \rightsquigarrow_< \{\alpha\}$ for every $\{\alpha\} \subseteq \mathcal{A} \setminus E$;
- *$<$-complete* if E is $<$-admissible and contains every assumption it $<$-defends;
- *$<$-preferred* if E is $\subseteq$-maximally $<$-admissible;
- *$<$-grounded* if E is $\subseteq$-minimally $<$-complete;
- *$<$-ideal* if E is $\subseteq$-maximal such that E is $<$-admissible and contained in all $<$-preferred extensions.

In Example 8, $\{\alpha\}$ is a unique $<$-stable, $<$-complete, $<$-preferred, $<$-grounded and $<$-ideal extension.

Henceforth, we assume $\sigma \in \{$stable, complete, preferred, grounded, ideal$\}$ and use $<$-σ to denote any ABA$^+$ semantics.

We recall several features that ABA$^+$ possesses and that will be used later.

Lemma 1. *Let $A' \subseteq A \subseteq \mathcal{A}$ and $B' \subseteq B \subseteq \mathcal{A}$ be given. If $A' \rightsquigarrow_< B'$, then $A \rightsquigarrow_< B$.*

Lemma 2. *For any $A, B \subseteq \mathcal{A}$:*

- *if $A \rightsquigarrow B$, then either $A \rightsquigarrow_< B$ or $B \rightsquigarrow_< A$;*
- *if $A \rightsquigarrow_< B$, then either $A \rightsquigarrow B$ or $B \rightsquigarrow A$.*

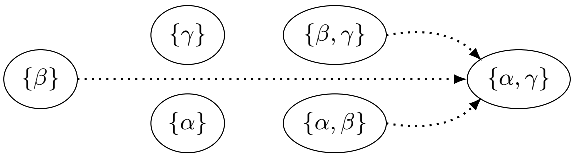
4 Properties of ABA$^+$ Semantics

To ensure that the familiar relations between semantics carry from ABA over to ABA$^+$, we want to guarantee the so-called Fundamental Lemma (Dung 1995; Bondarenko et al. 1997) (see below). To this end, we follow the well established structured argumentation formalism ASPIC$^+$ (Modgil and Prakken 2013; 2014) and impose the principle of *Contraposition*, reformulated for ABA$^+$ as follows.

Axiom 11. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ satisfies **the Axiom of Contraposition** if for all $A \subseteq \mathcal{A}$, $R \subseteq \mathcal{R}$ and $\beta \in \mathcal{A}$ it holds that if $A \vdash^R \overline{\beta}$, then for every $\alpha \in A$, there is $R_\alpha \subseteq \mathcal{R}$ with $(A \setminus \{\alpha\}) \cup \{\beta\} \vdash^{R_\alpha} \overline{\alpha}$.

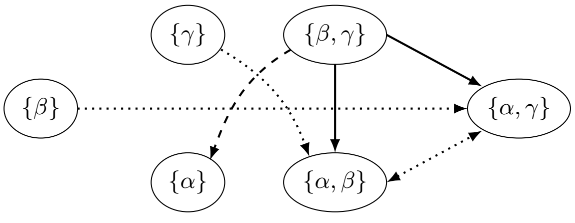
This axiom requires that if an assumption plays a role in deriving the contrary of another assumption, then it should contrapositively be possible for the latter to induce a derivation of the contrary of the former assumption too. The following example illustrates the effect Contraposition has in ABA$^+$.

Example 12. Let $\mathcal{R} = \{\overline{\beta} \leftarrow \alpha, \gamma\}$, $\mathcal{A} = \{\alpha, \beta, \gamma\}$ and $\alpha < \beta$, $\alpha < \gamma$. (The language and the contrary mapping are implicit from $\mathcal{R}$ and $\mathcal{A}$.) This ABA$^+$ framework $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ does not satisfy the Axiom of Contraposition. Its assumption framework (omitting $\emptyset$, $\mathcal{A}$ and $<$-attacks to and from $\mathcal{A}$) is shown below:



There are no extensions under, for instance, $<$-complete semantics, because all the singletons $\{\alpha\}$, $\{\beta\}$ and $\{\gamma\}$ are $<$-unattacked, but $\{\alpha, \beta, \gamma\}$ is not $<$-conflict-free.

If the rules $\overline{\alpha} \leftarrow \beta, \gamma$ and $\overline{\gamma} \leftarrow \alpha, \beta$ are added to $\mathcal{R}$ to constitute $\mathcal{R}'$, then the resulting $(\mathcal{L}, \mathcal{R}', \mathcal{A}, \overline{}, \leqslant)$ satisfies the Axiom of Contraposition and its assumption framework looks as follows ($<$-attacks that are both normal and reverse are depicted as solid directed edges):



Here, $\{\beta, \gamma\}$ is a unique $<$-complete extension.

We prove next that in the presence of Contraposition, the Fundamental Lemma is guaranteed to hold in ABA$^+$.

Lemma 3. *Suppose that $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ satisfies the Axiom of Contraposition. Let $S \subseteq \mathcal{A}$ be $<$-admissible and assume that S $<$-defends $\alpha, \alpha' \in \mathcal{A}$. Then $S \cup \{\alpha\}$ is $<$-admissible and $<$-defends α'.*

Proof. Note that if $\alpha \in S$, then $S \cup \{\alpha\}$ is trivially $<$-admissible. So assume $\alpha \notin S$ and suppose for a contradiction that $S \cup \{\alpha\}$ is not $<$-admissible. Then it is either not $<$-conflict-free, or does not $<$-defend itself. Suppose first $S \cup \{\alpha\} \rightsquigarrow_< S \cup \{\alpha\}$ via either (1) normal or (2) reverse attack. We show that either leads to a contradiction.

1. $S \cup \{\alpha\} \rightsquigarrow_< S \cup \{\alpha\}$ via normal attack. As S is $<$-conflict-free and $<$-defends α, this $<$-attack must involve α. I.e. $S' \cup \{\alpha\} \vdash^R \overline{\beta}$ for some $S' \subseteq S$ and $\beta \in S \cup \{\alpha\}$, and $\forall s' \in S' \cup \{\alpha\}$ we find $s' \not< \beta$. If $\beta = \alpha$, then $S' \cup \{\alpha\} \rightsquigarrow_< \{\alpha\}$, and so $S \rightsquigarrow_< S' \cup \{\alpha\}$. Else, if $\beta \in S'$, then

$S' \cup \{\alpha\} \rightsquigarrow_< S$, and so $S \rightsquigarrow_< S' \cup \{\alpha\}$ as well. We show that we can similarly obtain $S \rightsquigarrow_< S' \cup \{\alpha\}$ in case (2) too.

2. $S\cup\{\alpha\} \rightsquigarrow_< S\cup\{\alpha\}$ via reverse attack. As in 1., this $<$-attack must involve α, i.e. $S' \cup \{\alpha\} \vdash^{R} \overline{\beta}$ for some $S' \subseteq S$ and $\beta \in S \cup \{\alpha\}$, and $\exists s' \in S' \cup \{\alpha\}$ such that $s' < \beta$. If $\beta \in S$, then $S \rightsquigarrow_< S'\cup\{\alpha\}$. Else, if $\beta = \alpha$, then $s' \neq \alpha$ (by asymmetry of $<$), and using the Axiom of Contraposition we find $A \vdash^{R'} \overline{s'}$ for $A \subseteq (S'\cup\{\alpha\})\setminus\{s'\}$, so that $S'\cup\{\alpha\} \rightsquigarrow S$. Then, by Lemma 2, either $S' \cup \{\alpha\} \rightsquigarrow_< S$ or $S \rightsquigarrow_< S' \cup \{\alpha\}$, which yields $S \rightsquigarrow_< S' \cup \{\alpha\}$ in any case.

In either (1) or (2), $S \rightsquigarrow_< S' \cup \{\alpha\}$, and as S is $<$-conflict-free and $<$-defends α, this $<$-attack must be reverse and involve α: $A_1 \cup \{\alpha\} \vdash^{R_1} \overline{s_1}$, $s_1 \in S$, $A_1 \subseteq S'$, and $\exists s_1' \in A_1$ with $s_1' < s_1$. Without loss of generality take s_1' to be $\leqslant$-minimal such. By the Axiom of Contraposition, there is $S_1 \cup \{\alpha\} \vdash^{R_1'} \overline{s_1'}$ with $S_1 \subseteq (A_1 \setminus \{s_1'\}) \cup \{s_1\}$ and $\forall x \in S_1 \ x \not< s_1'$ (by $\leqslant$-minimality of s_1'). That is, $S_1 \cup \{\alpha\} \rightsquigarrow_< A_1$, so we find $S \rightsquigarrow_< S_1 \cup \{\alpha\}$, again via reverse attack involving α: $A_2 \cup \{\alpha\} \vdash^{R_2} \overline{s_2}$, $s_2 \in S$, $A_2 \subseteq S_1$, and $\exists s_2' \in A_2$ with $s_2' < s_2$. We again impose $\leqslant$-minimality on s_2' and by the Axiom of Contraposition get $S_2 \cup \{\alpha\} \vdash^{R_2'} \overline{s_2'}$, $S_2 \subseteq (A_2 \setminus \{s_2'\}) \cup \{s_2\}$ and $\forall x \in S_2 \ x \not< s_2'$.

As deductions are finite and $<$ asymmetric, the procedure described above will eventually exhaust pairs of $s_k' \in A_k$ and $s_k \in S_k$ such that $s_k' < s_k$, so that $S \rightsquigarrow_< S_k \cup \{\alpha\}$ will have to be a normal attack, for some k. But this leads to a contradiction to S being $<$-admissible and $<$-defending α.

Hence, by contradiction, $S \cup \{\alpha\}$ is $<$-conflict-free.

We now want to show that $S \cup \{\alpha\}$ $<$-defends itself. So let $B \rightsquigarrow_< S \cup \{\alpha\}$. As S is $<$-admissible and $<$-defends α, we consider this $<$-attack to be reverse and involving α: $S' \cup \{\alpha\} \vdash^{R} \overline{\beta_1}$, $S' \subseteq S$, $\beta_1 \in B$, and $\exists s' \in S' \cup \{\alpha\}$ with $s' < \beta_1$. By the Axiom of Contraposition, $S_1 \vdash^{R_1'} \overline{s'}$, $S_1 \subseteq ((S' \cup \{\alpha\}) \setminus \{s'\}) \cup \{\beta_1\}$. Thus, $S_1 \rightsquigarrow \{s'\}$, whence $S \cup \{\alpha\} \rightsquigarrow_< S_1$. This $<$-attack cannot be normal on $(S' \cup \{\alpha\}) \setminus \{s'\}$, due to $<$-conflict-freeness of $S \cup \{\alpha\}$; while, if it is normal on β_1, then $S\cup\{\alpha\} \rightsquigarrow_< B$, as required. Else, $S \cup \{\alpha\} \rightsquigarrow_< S_1$ via reverse attack: $B_1 \vdash^{R_1} \overline{s_1}$, $s_1 \in S \cup \{\alpha\}$, $B_1 \subseteq S_1$, and $\exists s_1' \in B_1$ with $s_1' < s_1$. Due to $<$-conflict-freeness of $S \cup \{\alpha\}$, we find $\beta_1 \in B_1$. Then again, by the Axiom of Contraposition, we find $S_2 \vdash^{R_2'} \overline{s_1'}$, $S_2 \subseteq (B_1 \setminus \{s_1'\}) \cup \{s_1\}$, and $\beta_1 \in S_2$. Like with the proof of $<$-conflict-freeness, this process must terminate with a normal attack $S \cup \{\alpha\} \rightsquigarrow_< B$, so that $S \cup \{\alpha\}$ eventually $<$-defends itself.

Finally, given that S $<$-defends α' to begin with, using Lemma 1 we conclude that $S \cup \{\alpha\}$ $<$-defends α' too. □

For the rest of this section, we assume that $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ satisfies the Axiom of Contraposition.

We can now define the $<$-defence operator *Def*, inspired by (Dung 1995).

Definition 13. *Def* : $\wp(\mathcal{A}) \to \wp(\mathcal{A})$ is defined as follows: for $A \subseteq \mathcal{A}$, $Def(A) = \{\alpha \in \mathcal{A} \ : \ A \ <\text{-defends } \alpha\}$.

By Lemma 1, *Def* is monotonic: if $A \subseteq B \subseteq \mathcal{A}$, then $Def(A) \subseteq Def(B)$. Hence, *Def* has a unique least fixed point, which is in addition a unique $<$-grounded extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, as shown next.

Proposition 4. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ *admits a unique $<$-grounded extension.*

Proof. First, observe that $\emptyset$ is $<$-admissible in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$. The least fixed point G can be given as $\bigcup_{i \in \mathbb{N}} Def^{\,i}(\emptyset)$. By Lemma 3, G is $<$-admissible. It is clearly $<$-complete (as $G = Def(G)$) and unique $\subseteq$-minimal such (as the least fixed point). Hence, G is a unique $<$-grounded extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$. □

As a consequence of Proposition 4, we get the following.

Corollary 5. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ *admits a $<$-complete extension.*

Using Lemma 3, we can prove the following results.

Proposition 6. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ *admits a $<$-preferred extension.*

Proof. By Lemma 3, the collection of $<$-admissible supersets of $\emptyset$ is partially ordered by subset inclusion $\subseteq$, so any sequence $\emptyset \subseteq A_1 \subseteq \ldots \subseteq A_n \subseteq \ldots$ of $<$-admissible sets of assumptions (for n an ordinal) has an upper bound $A = \bigcup_{i \geqslant 0} A_i$. Then $A \subseteq \mathcal{A}$ is $<$-admissible: if it were not $<$-conflict-free, then some A_n would not be either; and for any $B \rightsquigarrow_< A$ we have $B \rightsquigarrow_< A_n$, for some n, so that $A_n \rightsquigarrow_< B$, and hence $A \rightsquigarrow_< B$ too. Since every chain $\emptyset \subseteq A_1 \subseteq \ldots \subseteq A_n \subseteq \ldots$ admits an $<$-admissible upper bound, every such chain has a $\subseteq$-maximally $<$-admissible set of assumptions, according to Zorn's Lemma. As $\emptyset$ is $<$-admissible, $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ admits at least one $\subseteq$-maximally $<$-admissible—i.e. a $<$-preferred—extension. □

Proposition 7. *Every $<$-preferred extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is a $<$-complete extension too.*

Proof. Let E be a $<$-preferred extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ and suppose for a contradiction that it is not $<$-complete. Let E $<$-defend some $\alpha \in \mathcal{A} \setminus E$. As E is $<$-admissible, $E \cup \{\alpha\}$ is $<$-admissible, by Lemma 3. But then E is not $\subseteq$-maximally $<$-admissible, contrary to E being $<$-preferred. Hence, by contradiction, E must be $<$-complete. □

Further, as in ABA, $<$-stable semantics is subsumed by both $<$-preferred and $<$-complete semantics, as shown next.

Proposition 8. *Any $<$-stable extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is a $<$-preferred extension too.*

Proof. Let E be a $<$-stable extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$. As E $<$-attacks every $\{\beta\} \nsubseteq E$, it must be $\subseteq$-maximally $<$-admissible. Hence, E is $<$-preferred. □

Proposition 9. *Any $<$-stable extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is a $<$-complete extension too.*

Proof. Let E be a $<$-stable extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$. For any $\beta \notin E$, $<$-stability of E means that $E \rightsquigarrow_< \{\beta\}$, and if E $<$-defended β as well, it would mean that $E \rightsquigarrow_< E$, contradicting its $<$-conflict-freeness. Hence, E contains every assumption it $<$-defends, and so is $<$-complete. □

Finally, we consider $<$-ideal semantics.

Proposition 10. *$(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ admits a unique $<$-ideal extension.*

Proof. From Proposition 6 we know that $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ admits $<$-preferred extensions, so let S be their intersection. If $S = \emptyset$, then it is $<$-admissible, and so an $<$-ideal extension (unique). If $S \neq \emptyset$ is $<$-admissible, then it is an $<$-ideal extension (unique as well). Else, assume $S \neq \emptyset$ is not $<$-admissible. Then its $\subseteq$-maximally $<$-admissible subsets $I \subsetneq S$ are $<$-ideal extensions of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$. Suppose I and I' are two distinct $<$-admissible subsets of S. Then their union $I \cup I'$ is a subset of S too, and so $<$-conflict-free. By Lemma 3, $I \cup I'$ $<$-defends its assumptions, so must be $<$-admissible. Consequently, there can be only one $\subseteq$-maximally $<$-admissible subset of S, i.e. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ has a unique $<$-ideal extension. □

Proposition 11. *Any $<$-ideal extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is a $<$-complete extension too.*

Proof. By Proposition 10, it has a unique $<$-ideal extension I. Suppose for a contradiction that I is not $<$-complete. Then there is $\alpha \in \mathcal{A} \setminus I$ $<$-defended by I. Such α must be contained in the intersection S of $<$-preferred extensions of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, because $I \subseteq S$ $<$-defends α and every $<$-preferred extension E of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is $<$-complete (by Proposition 7). But then, $I \cup \{\alpha\}$ is $<$-admissible, according to Lemma 3, so that I is not $<$-ideal—a contradiction. Therefore, I must be $<$-complete. □

These properties that ABA$^+$ exhibits in the presence of Contraposition will be used to show, in the coming sections, that ABA$^+$ satisfies certain principles of preference handling and non-monotonic reasoning.

5 Preference Handling Properties

Referring to (Amgoud and Vesic 2009), in (Brewka, Truszczyński, and Woltran 2010) the authors hinted at two (arguably) desirable properties of argumentation formalisms dealing with preferences, that concern conflict preservation and the absence of preferences. In the next two subsections we indicate that ABA$^+$ satisfies those properties, and in the following subsections show that other (arguably) desirable properties of preference handling are too satisfied by ABA$^+$.

5.1 Conflict Preservation

The first property insists that extensions returned after accounting for preferences should be conflict-free with respect to attack relation not taking into account preferences. We formulate it as a principle applicable to ABA$^+$ as follows.

Definition 14. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ fulfils **the Principle of Conflict Preservation** for $<$-σ semantics if for all $<$-σ extensions $E \subseteq \mathcal{A}$ of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, for any $\alpha, \beta \in \mathcal{A}$, $\{\alpha\} \rightsquigarrow \{\beta\}$ implies that either $\alpha \notin E$ or $\beta \notin E$.

In (Čyras and Toni 2016) it was shown that Lemma 2 guarantees the following result.

Proposition 12. *$E \subseteq \mathcal{A}$ is conflict-free in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ iff E is $<$-conflict-free in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$.*

Consequently, ABA$^+$ ensures conflict preservation:

Proposition 13. *$(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ fulfils the Principle of Conflict Preservation for any semantics $<$-σ.*

Proof. Let E be a $<$-σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, If $\alpha, \beta \in E$ and $\{\alpha\} \rightsquigarrow \{\beta\}$, then $\{\alpha, \beta\}$ is not conflict-free, and hence not $<$-conflict-free, by Proposition 12. But then E is not $<$-conflict-free either, which is a contradiction. Thus, either one of α and β does not belong to E. □

5.2 Empty Preferences

The second property insists that if there are no preferences, then the extensions returned using a preference handling mechanism should be the same as those obtained without accounting for preferences. We formulate it as a principle applicable to ABA$^+$ as follows.

Definition 15. Suppose that the preference relation $\leqslant$ in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is the strict empty ordering $\emptyset$. Then $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \emptyset)$ fulfils **the Principle of Empty Preferences** for $\emptyset$-σ semantics if for all $\emptyset$-σ extensions $E \subseteq \mathcal{A}$ of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \emptyset)$, E is a σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$.

In (Čyras and Toni 2016) the following result was shown to hold.

Theorem 14. *$E \subseteq \mathcal{A}$ is a σ-extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ iff E is an $\emptyset$-σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \emptyset)$.*

This theorem, in addition to saying that ABA$^+$ is a conservative extension of ABA, immediately yields the satisfaction of the principle in question:

Proposition 15. *$(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \emptyset)$ fulfils the Principle of Empty Preferences for any semantics $\emptyset$-σ.*

5.3 Maximal Elements

(Amgoud and Vesic 2014) proposed a property concerning inclusion in extensions of the 'strongest' arguments, i.e. arguments that are maximal w.r.t. preference ordering. We next reformulate the property to be applicable to ABA$^+$.

Definition 16. Suppose the preference ordering $\leqslant$ of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ is total and further assume that the set $M = \{\alpha \in \mathcal{A} : \nexists \beta \in \mathcal{A} \text{ with } \alpha < \beta\}$ is $<$-conflict-free. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ fulfils **the Principle of Maximal Elements** for $<$-σ semantics if for all $<$-σ extensions $E \subseteq \mathcal{A}$ of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, it holds that $M \subseteq E$.

As an illustration, in Example 8, α is a unique $\leqslant$-maximal element in $\mathcal{A}$, and $\{\alpha\}$ is a unique $<$-σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, whence $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ fulfils the Principle of Maximal Elements for any semantics $<$-σ.

Our next result shows that in general, ABA$^+$ satisfies this principle under $<$-stable and $<$-complete semantics.

Proposition 16. *$(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ fulfils the Principle of Maximal Elements for $<$-stable and $<$-complete semantics.*

Proof. Let the preference ordering $\leqslant$ of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ be total and suppose $M = \{\alpha \in \mathcal{A} : \nexists \beta \in \mathcal{A} \text{ with } \alpha < \beta\}$ is $<$-conflict-free. We first show that M is not $<$-attacked.

Fix $\alpha \in M$ and suppose for a contradiction that for some $S \subseteq \mathcal{A}$ it holds that $S \rightsquigarrow_< \{\alpha\}$. So either (i) $\exists B \vdash^R \overline{\alpha}$ with

$B \subseteq S$ and $\forall \beta \in B$ $\alpha \leqslant \beta$ or $\beta \nleqslant \alpha$, or (ii) $\{\alpha\} \vdash^R \overline{\beta}$ for some $\beta \in S$ with $\alpha < \beta$. Note that the case (ii) cannot happen, because α is $\leqslant$-maximal. So consider case (i). Since $\leqslant$ is total, it follows that $\alpha \leqslant \beta$ $\forall \beta \in B$. But as α is $\leqslant$-maximal, it must also hold that $\beta \leqslant \alpha$, for any $\beta \in B$. From here, we show $B \subseteq M$. Indeed, fix $\beta \in B$ and assume for a contradiction that $\beta \notin M$. Then $\exists \gamma \in \mathcal{A}$ such that $\beta < \gamma$. By transitivity, $\alpha < \gamma$, contradicting α's $\leqslant$-maximality. So we must have $\beta \in M$, and consequently, $B \subseteq M$.

But now, since $\alpha \in M$, $B \subseteq M$ and $B \leadsto_< \{\alpha\}$, this contradicts $<$-conflict-freeness of M. Therefore, by contradiction, $S \not\leadsto_< \{\alpha\}$, for any $S \subseteq \mathcal{A}$. Since $\alpha \in M$ was arbitrary, we have M $<$-unattacked, as required.

If $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ admits no $<$-stable or $<$-complete extensions, then the principle is fulfilled trivially. Otherwise, let $E \subseteq \mathcal{A}$ be $<$-stable in $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$. Pick $\alpha \in M$ and suppose for a contradiction that $\alpha \notin E$. Then $E \leadsto_< \{\alpha\}$, which is a contradiction. Thus, $\alpha \in S$, and hence $M \subseteq S$.

Now let E be a $<$-complete extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ and suppose for a contradiction $M \nsubseteq E$. Then E does not $<$-defend some $\alpha \in M$. This means that $S \leadsto_< M$ for some $S \subseteq \mathcal{A}$, which is a contradiction. Hence, $M \subseteq E$. □

This principle may, however, be violated under, say, $<$-preferred semantics: in Example 12, the framework $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ to begin with, admits $\{\alpha, \beta\}$ as a $<$-preferred extension, while $\gamma \notin \{\alpha, \beta\}$ is a $\leqslant$-maximal element. However, assuming Contraposition, the Principle of Maximal Elements is satisfied under the remaining semantics too.

Corollary 17. *If $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ satisfies the Axiom of Contraposition, then it fulfils the Principle of Maximal Elements for $<$-preferred/$<$-ideal/$<$-grounded semantics.*

Proof. Follows from Propositions 4, 7, 11 and 16. □

5.4 Principle I

(Brewka and Eiter 2000) formulated a principle for sound extension-based default reasoning with preferences, which we reformulate for ABA$^+$ next.

Definition 17. $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ fulfils **Principle I** for $<$-σ semantics if for all $E, E' \subseteq \mathcal{A}$ such that $E = E_0 \cup \{\alpha\}$ and $E' = E_0 \cup \{\alpha'\}$ for some $E_0 \subseteq \mathcal{A}$, with $\alpha, \alpha' \notin E_0$ and $\alpha' < \alpha$, it holds that if E is a $<$-σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$, then E' is not a $<$-σ extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$.

This principle insists that if two coherent viewpoints of a situation differ only in that each of them contains a single assumption not contained in the other, then the viewpoint with the more preferred assumption should be chosen. ABA$^+$ satisfies this principle under $<$-stable semantics.

Proposition 18. *$(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ fulfils Principle I for $<$-stable semantics.*

Proof. Suppose for a contradiction that both $E = E_0 \cup \{\alpha\}$ and $E' = E_0 \cup \{\alpha'\}$, where $\alpha' < \alpha$, are $<$-stable extensions of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$. As E' is $<$-stable and $\alpha \notin E'$, we get $E' \leadsto_< \{\alpha\}$. As E is $<$-conflict-free, we find $E_0 \not\leadsto_< \{\alpha\}$, so (from $E' \leadsto_< \{\alpha\}$ we get that): (i) either there is $E'' \cup \{\alpha'\} \vdash^R \overline{\alpha}$ with $E'' \subseteq E_0$ and $\varepsilon \nless \alpha$ $\forall \varepsilon \in E'' \cup \{\alpha'\}$; (ii) or $\{\alpha\} \vdash^R \overline{\alpha'}$ is such that $\alpha < \alpha'$. As $\alpha' < \alpha$, both cases lead to a contradiction, so that E' is not a $<$-stable extension, provided E is. □

In Example 8, $E = \{\alpha\}$ is a unique $<$-stable extension of $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$, which illustrates the principle as follows: take $E_0 = \emptyset$ so that $E = \{\alpha\}$ and $E' = \{\beta\}$, where $\beta < \alpha$. It is important that Principle I is satisfied under $<$-stable semantics, because (Brewka and Eiter 1999) investigated (preferred) answer sets of logic programs, and answer sets in Logic Programming correspond to stable extensions in ABA (Bondarenko et al. 1997). Satisfaction of the principle gives hope that preferred answer set semantics can be captured in ABA$^+$, as answer set semantics is captured in ABA.

Principle I, however, may be violated under $<$-preferred semantics: in Example 12, $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-, \leqslant)$ has two $<$-preferred extensions $\{\alpha, \beta\}$ and $\{\beta, \gamma\}$, and yet $\alpha < \gamma$. Note, though, that $(\mathcal{L}, \mathcal{R}', \mathcal{A}, ^-, \leqslant)$ satisfies the Axiom of Contraposition and has a unique $<$-σ extension $\{\beta, \gamma\}$, and thus fulfils Principle I for any semantics $<$-σ. Based on our investigations, we conjecture that assuming Contraposition, ABA$^+$ frameworks fulfil the principle for the remaining semantics as well. Verifying this is left as future work.

6 Non-Monotonic Reasoning Properties

(Čyras and Toni 2015) proposed and studied the well known non-monotonic inference properties of *Cautious Monotonicity* (MON henceforth) and *Cumulative Transitivity* (CUT henceforth) for ABA. Here, we investigate some of those properties for ABA$^+$. We first recall (some of) the properties considered and results obtained.[1]

Assume as given a fixed, but otherwise arbitrary (flat) ABA framework $\mathcal{F} = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-)$. Let E be a σ extension of $\mathcal{F}$. In what follows, E' will denote a σ extension of a newly constructed ABA framework $\mathcal{F}'$. To avoid trivialities, we consider cases only where each of $\mathcal{F}$ and $\mathcal{F}'$ has at least one σ extension—E and E' respectively.

We first recall the STRICT setting regarding *strengthening of information*. Given $\psi \in Cn(E) \setminus \mathcal{A}$, define $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A}, ^-)$. There are four properties:

SCEPTICAL STRICT CUT :

For all extensions E' of $\mathcal{F}'$ we have $Cn(E') \subseteq Cn(E)$;

CREDULOUS STRICT CUT :

There is an extension E' of $\mathcal{F}'$ with $Cn(E') \subseteq Cn(E)$;

SCEPTICAL STRICT MON :

For all extensions E' of $\mathcal{F}'$ we have $Cn(E) \subseteq Cn(E')$;

CREDULOUS STRICT MON :

There is an extension E' of $\mathcal{F}'$ with $Cn(E) \subseteq Cn(E')$.

Table 1 summarizes results pertaining to ABA (sceptical and credulous versions coincide under grounded and ideal

[1] In (Čyras and Toni 2015), instead of sceptical/credulous (see below) the words strong/weak were used, respectively; we have altered the names to adhere to the more common terminology.

semantics, and for other semantics the status of the credulous property is indicated in parentheses).

Property	Grd.	Ideal	Stable	Pref.	Cpl.
STRICT CUT	✓	✓	X (✓)	X (✓)	X (✓)
STRICT MON	✓	X	X (✓)	X (✓)	X (✓)

Table 1: STRICT CUT / MON for standard ABA

We now recall the ASM setting, where conclusions that are themselves assumptions are being *confirmed*. Given $\psi \in Cn(E) \cap \mathcal{A}$, define $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A} \setminus \{\psi\}, \overline{})$.[2] The properties are as follows:

SCEPTICAL ASM CUT :

For all extensions E' of $\mathcal{F}'$ we have $Cn(E') \subseteq Cn(E)$;

CREDULOUS ASM CUT :

There is an extension E' of $\mathcal{F}'$ with $Cn(E') \subseteq Cn(E)$;

SCEPTICAL ASM MON :

For all extensions E' of $\mathcal{F}'$ we have $Cn(E) \subseteq Cn(E')$;

CREDULOUS ASM MON :

There is an extension E' of $\mathcal{F}'$ with $Cn(E) \subseteq Cn(E')$.

Table 2 summarizes results regarding ABA in the ASM setting (notation as before).

Property	Grd.	Ideal	Stable	Pref.	Cpl.
ASM CUT	✓	✓	X (✓)	X (✓)	X (✓)
ASM MON	✓	X	X (✓)	X (✓)	X (✓)

Table 2: ASM CUT / MON for standard ABA

The non-monotonic inference properties CUT and MON can be readily applied to ABA$^+$. Take $\mathcal{F}$ to be an ABA$^+$ framework $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, let E be its $<$-σ extension, and given $\psi \in Cn(E)$, define $\mathcal{F}'$ as follows:

- STRICT setting: $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A}, \overline{}, \leqslant)$;
- ASM setting: $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A} \setminus \{\psi\}, \overline{}, \leqslant')$, where $\leqslant'$ is a restriction of $\leqslant$ to $\mathcal{A} \setminus \{\psi\}$.

We can then analyse whether the non-monotonic inference properties in question are satisfied in ABA$^+$. Trivially, as ABA$^+$ is a conservative extension of ABA (cf. Theorem 14), properties violated in ABA will remain violated in ABA$^+$. Therefore, we will focus on those that are satisfied in ABA; in particular, the credulous versions except for MON under ideal semantics.

Example 18. As an illustration of the properties, recall Example 12. The ABA$^+$ framework $\mathcal{F} = (\mathcal{L}, \mathcal{R}', \mathcal{A}, \overline{}, \leqslant)$ (that satisfies the Axiom of Contraposition) has a unique $<$-σ extension $\{\beta, \gamma\}$ with $Cn(\{\beta, \gamma\}) = \{\overline{\alpha}, \beta, \gamma\}$.

- STRICT setting: take $\overline{\alpha}$ and let $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\overline{\alpha} \leftarrow \top\}, \mathcal{A}, \overline{}, \leqslant)$. Then $\mathcal{F}'$ has a unique $<$-σ extension $\{\beta, \gamma\}$.
- ASM setting: take β and let $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\beta \leftarrow \top\}, \mathcal{A} \setminus \{\beta\}, \overline{}, \leqslant')$ with $\alpha <' \gamma$. Then $\mathcal{F}'$ likewise has a unique $<$-σ extension $\{\beta, \gamma\}$.

As conclusions of extensions of both $\mathcal{F}$ and $\mathcal{F}'$ are actually the same, the credulous versions of the properties are indeed satisfied in both settings.

In what follows, we assume that $(\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$ satisfies the Axiom of Contraposition and show that ABA$^+$ retains the same satisfaction results of CUT and MON from ABA in both STRICT and ASM settings.

Proposition 19. *$<$-complete semantics satisfies CREDULOUS STRICT CUT and CREDULOUS STRICT MON.*

Proof. Let E be a $<$-complete extension of $\mathcal{F} = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \leqslant)$, fix $\psi \in Cn(E) \setminus \mathcal{A}$, and let $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A}, \overline{}, \leqslant)$. For ease of reference, we will denote by $\rightsquigarrow_<$ and $\rightsquigarrow'_<$ the $<$-attack relations in $\mathcal{F}$ and $\mathcal{F}'$ respectively. We claim that E is a $<$-complete extension of $\mathcal{F}'$ too. First, E is clearly $<$-conflict-free in $\mathcal{F}'$. Second, let $\alpha \in E$ and suppose that $B' \rightsquigarrow'_< \{\alpha\}$ for some $B' \subseteq \mathcal{A}$ with $B' \setminus E$. There are two possibilities.

Possibility 1: this $<$-attack uses the rule $\psi \leftarrow \top$. We split into cases.

- First, assume $B' \rightsquigarrow'_< \{\alpha\}$ via normal attack. I.e., $\exists B \vdash^R \overline{\alpha}$ with $B \subseteq B'$, $R \subseteq \mathcal{R}$, and such that $\forall \beta \in B \ \ \beta \nless \alpha$. Consider some $E_0 \subseteq E$ with $E_0 \vdash^{R_0} \psi$, for some $R_0 \subseteq \mathcal{R}$. We have $B \cup E_0 \vdash^{R \cup R_0} \overline{\alpha}$.
  - If $\forall \varepsilon \in E_0$ we have $\varepsilon \nless \alpha$, then $B \cup E_0 \rightsquigarrow_< \{\alpha\}$, so that $E \rightsquigarrow_< B \cup E_0$, and thus (as E is $<$-admissible in $\mathcal{F}$ and $E_0 \subseteq E$) we find $E \rightsquigarrow_< B$, whence $E \rightsquigarrow'_< B$ as well. Thus, $E \rightsquigarrow_< B'$, as required.
  - Else, if $\exists \varepsilon \in E_0$ with $\varepsilon < \alpha$, take $\leqslant$-minimal such. Then by the Axiom of Contraposition, there is $(B \cup E_0 \setminus \{\varepsilon\}) \cup \{\alpha\} \vdash^{R'} \overline{\varepsilon}$, and by $\leqslant$-minimality of ε, we find that $\nexists x \in (B \cup E_0 \setminus \{\varepsilon\}) \cup \{\alpha\}$ such that $x < \varepsilon$. Hence, $(B \cup E_0 \setminus \{\varepsilon\}) \cup \{\alpha\} \rightsquigarrow_< \{\varepsilon\}$, so that $E \rightsquigarrow_< B \cup E$, and hence $E \rightsquigarrow'_< B$, as in the previous case.
- Now assume $B' \rightsquigarrow'_< \{\alpha\}$ is a reverse attack, i.e., $\{\alpha\} \vdash^R \overline{\beta}$, $\beta \in B'$, $R \subseteq \mathcal{R}$ and $\alpha < \beta$. By the Axiom of Contraposition, $\{\beta\} \vdash^{R'} \overline{\alpha}$ via normal attack. Hence, we are back in the first case above.

In any case, E $<$-defends α in $\mathcal{F}'$.

Possibility 2: the $<$-attack $B' \rightsquigarrow'_< \{\alpha\}$ does not involve the rule $\psi \leftarrow \top$. That is, we actually have $B' \rightsquigarrow_< \{\alpha\}$. Then, $E \rightsquigarrow_< B'$, and hence $E \rightsquigarrow'_< B'$.

In any event, E $<$-defends α in $\mathcal{F}'$. Since $\alpha \in E$ was arbitrary, we conclude that E is $<$-admissible in $\mathcal{F}'$.

It now suffices to show that E contains every assumption it $<$-defends in $\mathcal{F}'$. To this end, suppose E $<$-defends α in $\mathcal{F}'$, and suppose for a contradiction that $\alpha \notin E$. Then E does not $<$-defend α in $\mathcal{F}$. That is, there is $B \rightsquigarrow_< \{\alpha\}$ such that $E \not\rightsquigarrow_< B$. But now, we also have $B \rightsquigarrow'_< \{\alpha\}$, so that $E \rightsquigarrow'_< B$, whence it must be that $E \rightsquigarrow'_< B$ is a normal attack that does not use some assumption $\varepsilon \in E$ (which is used to deduce ψ, i.e. $E_0 \vdash^{R_0} \psi$, $\varepsilon \in E_0 \subseteq E$,

[2] For brevity reasons, the same symbol $\overline{}$ is used for both contrary mappings, and in the new framework $\mathcal{F}'$, the contrary mapping $\overline{}$ is implicitly restricted to a diminished set of assumptions.

$R_0 \subseteq \mathcal{R}$) such that $\varepsilon < \beta$ for some $\beta \in B$. Taking $\leqslant$-minimal such ε (and accordingly some $\beta in B$), the Axiom of Contraposition guarantees that $(E \setminus \{\varepsilon\}) \cup \{\beta\} \rightsquigarrow_< \{\varepsilon\}$ via normal attack, and since $\varepsilon \in E$, it must be that $E \rightsquigarrow_< \{\beta\}$, giving $E \rightsquigarrow_< B$, which is a contradiction. Hence, $\alpha \in E$ after all, and so E is $<$-complete in $\mathcal{F}'$, as required. □

Proposition 20. *$<$-preferred semantics satisfies* `CREDULOUS STRICT CUT` *and* `CREDULOUS STRICT MON`*.*

Proof. Like in the proof of Proposition 19, we claim that a $<$-preferred extension E of $\mathcal{F}$ is a $<$-preferred extension of $\mathcal{F}'$. Indeed, if E were not $\subseteq$-maximally $<$-admissible in $\mathcal{F}'$, then for some $\beta \in \mathcal{A} \setminus E$, $E \cup \{\beta\}$ would be $<$-admissible in $\mathcal{F}'$. Verbatim to the proof for $<$-complete semantics, we could show that $E \cup \{\beta\}$ is $<$-admissible in $\mathcal{F}$ too, contradicting E being $<$-preferred in $\mathcal{F}$. □

Proposition 21. *$<$-stable semantics satisfies* `CREDULOUS STRICT CUT` *and* `CREDULOUS STRICT MON`*.*

Proof. Like in the proof of Proposition 19, we claim that a $<$-stable extension E of $\mathcal{F}$ is a $<$-stable extension of $\mathcal{F}'$. Indeed, let $\beta \notin E$. Then $E \rightsquigarrow_< \{\beta\}$. Whether it is a normal or reverse attack, we clearly have $E \rightsquigarrow'_< \{\beta\}$ too. Hence, E is $<$-stable in $\mathcal{F}'$, provided E is $<$-stable in $\mathcal{F}$. □

Proposition 22. *$<$-grounded semantics satisfies* `CREDULOUS STRICT CUT` *and* `CREDULOUS STRICT MON`*.*

Sketch. Using the argument as in Proposition 19, it can be proven by induction on the construction of the $<$-grounded extension G of $\mathcal{F}$ (cf. Proposition 4) that G is the $<$-grounded extension of $\mathcal{F}'$. □

Corollary 23. *$<$-ideal semantics satisfies* `CREDULOUS STRICT CUT`*.*

Proof. This follows by definition of the $<$-ideal extension and Proposition 20. □

Now, in the `ASM` setting, the came results can be obtained as in the `STRICT` setting, with essentially the same proofs.

Proposition 24. *$<$-complete/$<$-preferred/$<$-stable/$<$-grounded semantics satisfies* `CREDULOUS ASM CUT` *and* `CREDULOUS ASM MON`*, and $<$-ideal semantics satisfies* `CREDULOUS ASM CUT`*.*

Sketch. Let E be a $<$-complete extension of $\mathcal{F} = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-}, \leqslant)$, fix $\psi \in Cn(E) \cap \mathcal{A}$, and let $\mathcal{F}' = (\mathcal{L}, \mathcal{R} \cup \{\psi \leftarrow \top\}, \mathcal{A} \setminus \{\psi\}, ^{-}, \leqslant')$, where $\leqslant'$ is a restriction of $\leqslant$ to $\mathcal{A} \setminus \{\psi\}$. It can be shown that E is a $<$-complete extension of $\mathcal{F}'$, by replacing, in the proof of Proposition 19, E_0 and $E_0 \vdash^{R_0} \psi$ with $\{\psi\}$ and $\{\psi\} \vdash^{\emptyset} \psi$ respectively. Other claims follow the same line of reasoning as for the proofs in the `STRICT` setting. □

Table 3 summarizes this section's results (sceptical and credulous versions coincide under $<$-grounded and $<$-ideal semantics; for other semantics the credulous version is indicated in parentheses.)

Property	$<$-g.	$<$-id.	$<$-stb.	$<$-pr.	$<$-cpl.
`STRICT` and `ASM CUT`	✓	✓	X (✓)	X (✓)	X (✓)
`STRICT` and `ASM MON`	✓	X	X (✓)	X (✓)	X (✓)

Table 3: (`STRICT` and `ASM`) `CUT` and `MON` for ABA$^+$

7 Related and Future Work

The principle of Contraposition of (strict) rules (see e.g. (Caminada and Amgoud 2007; Modgil and Prakken 2013)) is notably employed in the well studied structured argumentation formalism ASPIC$^+$ (Modgil and Prakken 2013; 2014). The principle as such is also inherently present in classical logic-based approaches to structured argumentation such as (Gorogiannis and Hunter 2011; Besnard and Hunter 2014). Similarly as in ASPIC$^+$, ABA$^+$ utilizes Contraposition to ensure the Fundamental Lemma (cf. Lemma 3). As a consequence, Contraposition paves way to satisfaction of desirable properties of ABA$^+$ semantics, as well as preference handling and non-monotonic inference properties discussed in Sections 5 and 6. Whether the Axiom of Contraposition can be relaxed for ABA$^+$ to obtain the same results is a line of future research.

The preference handling principle discussed in Section 5.4 was originally proposed, along with some other properties, by (Brewka and Eiter 1999) for answer set programming (ASP) with preferences. To the best of our knowledge, reformulation of Principle I for ABA$^+$ is the first application of this principle to argumentation with preferences. Building on (Brewka and Eiter 1999), (Šimko 2014) discussed an extended set of principles for ASP with preferences, most of which focus on preferences over rules. Whether those principles can be applied to ABA$^+$ is a future work direction.

Regarding preference handling in argumentation, along with the Principle of Maximal Elements discussed in Section 5.3, (Amgoud and Vesic 2014) suggested several arguably desirable properties of argumentation with preferences. Those properties are exhibited in ABA$^+$ as Proposition 12 and Theorem 14. Referring to those properties, (Brewka, Truszczyński, and Woltran 2010) also hinted at other properties regarding selection among extensions, as possible principles of preference handling in argumentation. Relating those principles to ABA$^+$ is left for future work.

In terms on non-monotonic reasoning properties, Cautious Monotonicity and Cumulative Transitivity (studied in Section 6) are traced to (Makinson 1988; Kraus, Lehmann, and Magidor 1990) and fall into the well studied area of analysing non-monotonic reasoning with respect to information change (cf. (Rott 2001)). In argumentation setting, the latter is also known as *argumentation dynamics*, and has recently been a topic of interest in the argumentation community (see e.g. (Cayrol, de Saint-Cyr, and Lagasquie-Schiex 2010; Falappa et al. 2011; Baroni et al. 2014; Coste-Marquis et al. 2014; Booth et al. 2014; Diller et al. 2015; Baumann and Brewka 2015)). In particular, non-monotonic inference properties were investigated in (Hunter 2010) with respect to argument–claim entailment in logic-based argumentation systems; in (Čyras and Toni 2015) for ABA; and with regards to ASPIC$^+$-type-of argumentation systems in (Dung 2016). Only the latter of the three works concerns argumentation with preferences. In addition to considering different structured argumentation setting and different preference handling mechanisms, it diverges from our analysis in Section 6 in that (Dung 2016) regards Cumulative Transitivity plus Cautious Monotonicity as a single property of Cumulativity and studies it only for stable and complete semantics. Other argumentation-related properties from (Dung 2016) will be studied for ABA$^+$ in the future.

Several other topics of interest are left for future work. For instance, integrating *dynamic preferences* (see e.g. (Prakken and Sartor 1999; Zhang and Foo 1997; Brewka and Woltran 2010)) within ABA$^+$ and studying their interaction with the properties of preference handling as well as of non-monotonic inference. Also, relation of ABA$^+$ to Logic Programming with preferences (e.g. (Sakama and Inoue 1996; Zhang and Foo 1997; Brewka and Eiter 1999)) and non-monotonic reasoning formalisms equipped with preferences in general (e.g. (Brewka 1989; Baader and Hollunder 1995; Rintanen 1998; Brewka and Eiter 2000; Delgrande and Schaub 2000; Stolzenburg et al. 2003; Kakas and Moraitis 2003)) is left for future research.

There are as well numerous approaches to integrating reasoning with preferences within argumentation, e.g. (Amgoud and Cayrol 2002; Bench-Capon 2003; Kaci and van der Torre 2008; Modgil 2009; Modgil and Prakken 2010; Baroni et al. 2011; Dunne et al. 2011; Brewka et al. 2013; Amgoud and Vesic 2014; Besnard and Hunter 2014; García and Simari 2014; Wakaki 2014; Modgil and Prakken 2013; 2014; Dung 2016). It would be interesting to study these formalisms with respect to the properties considered in this paper, where it has not already been done. We leave this as future work.

8 Conclusions

We investigated various properties of a recently proposed non-monotonic reasoning formalism ABA$^+$ (Čyras and Toni 2016) that deals with preferences in structured argumentation. In particular, we first established that assuming the principle of Contraposition (see e.g. (Modgil and Prakken 2013)), ABA$^+$ semantics exhibit desirable properties akin to those of other existing argumentation formalisms, such as (Dung 1995). We then showed that ABA$^+$ satisfies some (arguably) desirable principles of preference handling in argumentation and non-monotonic reasoning, e.g. (Brewka and Eiter 1999). Finally, we analysed non-monotonic inference properties (as in (Čyras and Toni 2015)) of ABA$^+$ under various semantics. We believe our work contributes to the understanding of preferences within argumentation in particular, and in non-monotonic reasoning at large.

References

Amgoud, L., and Cayrol, C. 2002. A Reasoning Model Based on the Production of Acceptable Arguments. *Ann. Math. Artif. Intell.* 34(1-3):197–215.

Amgoud, L., and Vesic, S. 2009. Repairing Preference-Based Argumentation Frameworks. In *IJCAI*, 665–670.

Amgoud, L., and Vesic, S. 2014. Rich Preference-Based Argumentation Frameworks. *Int. J. Approx. Reason.* 55(2):585–606.

Baader, F., and Hollunder, B. 1995. Priorities on Defaults with Prerequisites, and Their Application in Treating Specificity in Terminological Default Logic. *J. Autom. Reason.* 15(1):41–68.

Baroni, P.; Cerutti, F.; Giacomin, M.; and Guida, G. 2011. AFRA: Argumentation Framework with Recursive Attacks. *Int. J. Approx. Reason.* 52(1):19–37.

Baroni, P.; Boella, G.; Cerutti, F.; Giacomin, M.; van der Torre, L.; and Villata, S. 2014. On the Input/Output Behavior of Argumentation Frameworks. *Artif. Intell.* 217:144–197.

Baumann, R., and Brewka, G. 2015. AGM Meets Abstract Argumentation: Expansion and Revision for Dung Frameworks. In *IJCAI*, 2734–2740.

Bench-Capon, T. 2003. Persuasion in Practical Argument Using Value Based Argumentation Frameworks. *J. Log. Comput.* 13(3):429–448.

Besnard, P., and Hunter, A. 2014. Constructing Argument Graphs with Deductive Arguments: A Tutorial. *Argum.&Comput.* 5(1):5–30.

Besnard, P.; García, A. J.; Hunter, A.; Modgil, S.; Prakken, H.; Simari, G. R.; and Toni, F. 2014. Introduction to Structured Argumentation. *Argum.&Comput.* 5(1):1–4.

Bondarenko, A.; Dung, P. M.; Kowalski, R.; and Toni, F. 1997. An Abstract, Argumentation-Theoretic Approach to Default Reasoning. *Artif. Intell.* 93(97):63–101.

Booth, R.; Gabbay, D.; Kaci, S.; Rienstra, T.; and van der Torre, L. 2014. Abduction and Dialogical Proof in Argumentation and Logic Programming. In *ECAI*.

Brewka, G., and Eiter, T. 1999. Preferred Answer Sets for Extended Logic Programs. *Artif. Intell.* 109(1-2):297–356.

Brewka, G., and Eiter, T. 2000. Prioritizing Default Logic. In *Intellectics Comput. Log.*, 27–45.

Brewka, G., and Woltran, S. 2010. Abstract Dialectical Frameworks. In *KR*.

Brewka, G.; Ellmauthaler, S.; Strass, H.; Wallner, J.; and Woltran, S. 2013. Abstract Dialectical Frameworks Revisited. In *IJCAI*, 803–809.

Brewka, G.; Niemelä, I.; and Truszczyński, M. 2007. Non-monotonic reasoning. In van Harmelen, F.; Lifschitz, V.; and Bruce, P., eds., *Handb. Knowl. Represent.* Elsevier. 239–284.

Brewka, G.; Truszczyński, M.; and Niemelä, I. 2008. Preferences and Nonmonotonic Reasoning. *AI Mag.* 29(4):69–78.

Brewka, G.; Truszczyński, M.; and Woltran, S. 2010. Representing Preferences Among Sets. In *AAAI*, 273–278.

Brewka, G. 1989. Preferred Subtheories: An Extended Logical Framework for Default Reasoning. In *IJCAI*, 1043–1048.

Caminada, M., and Amgoud, L. 2007. On the Evaluation of Argumentation Formalisms. *Artif. Intell.* 171(5-6):286–310.

Cayrol, C.; de Saint-Cyr, F.; and Lagasquie-Schiex, M.-C. 2010. Change in Abstract Argumentation Frameworks: Adding an Argument. *J. Artif. Intell. Res.* 38(1):49–84.

Coste-Marquis, S.; Konieczny, S.; Mailly, J.-G.; and Marquis, P. 2014. On the Revision of Argumentation Systems: Minimal Change of Arguments Status. In *KR*.

Čyras, K., and Toni, F. 2015. Non-Monotonic Inference Properties for Assumption-Based Argumentation. In *TAFA*, 92–111.

Čyras, K., and Toni, F. 2016. ABA+: Assumption-Based Argumentation with Preferences. In *KR*.

Delgrande, J., and Schaub, T. 2000. Expressing Preferences in Default Logic. *Artif. Intell.* 123(1-2):41–87.

Delgrande, J.; Schaub, T.; Tompits, H.; and Wang, K. 2004. A Classification and Survey of Preference Handling Approaches in Nonmonotonic Reasoning. *Comput. Intell.* 20(2):308–334.

Diller, M.; Haret, A.; Linsbichler, T.; Rummele, S.; and Woltran, S. 2015. An Extension-Based Approach to Belief Revision in Abstract Argumentation. In *IJCAI*, 2926–2932.

Domshlak, C.; Hüllermeier, E.; Kaci, S.; and Prade, H. 2011. Preferences in AI: An Overview. *Artif. Intell.* 175(7-8):1037–1052.

Dung, P. M. 1995. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-person Games. *Artif. Intell.* 77:321–357.

Dung, P. M. 2016. An axiomatic Analysis of Structured Argumentation with Priorities. *Artif. Intell.* 231:107–150.

Dunne, P.; Hunter, A.; McBurney, P.; Parsons, S.; and Wooldridge, M. 2011. Weighted Argument Systems: Basic Definitions, Algorithms, and Complexity Results. *Artif. Intell.* 175(2):457–486.

Falappa, M.; García, A. J.; Kern-Isberner, G.; and Simari, G. R. 2011. On the Evolving Relation between Belief Revision and Argumentation. *Knowl. Eng. Rev.* 26(01):35–43.

García, A. J., and Simari, G. R. 2014. Defeasible Logic Programming: DeLP-servers, Contextual Queries, and Explanations for Answers. *Argum.&Comput.* 5(1):63–88.

Gorogiannis, N., and Hunter, A. 2011. Instantiating Abstract Argumentation with Classical Logic Arguments: Postulates and Properties. *Artif. Intell.* 175(9-10):1479–1497.

Hunter, A. 2010. Base Logics in Argumentation. In *COMMA*, 275–286.

Kaci, S., and van der Torre, L. 2008. Preference-Based Argumentation: Arguments Supporting Multiple Values. *Int. J. Approx. Reason.* 48(3):730–751.

Kaci, S. 2011. *Working with Preferences. Less is More*. Springer.

Kakas, A., and Moraitis, P. 2003. Argumentation Based Decision Making for Autonomous Agents. In *AAMAS*, 883–890.

Kraus, S.; Lehmann, D.; and Magidor, M. 1990. Nonmonotonic Reasoning, Preferential Models and Cumulative Logics. *Artif. Intell.* 44(1-2):167–207.

Makinson, D. 1988. General Theory of Cumulative Inference. In *NMR*, 1–18.

Modgil, S., and Prakken, H. 2010. Reasoning About Preferences in Structured Extended Argumentation Frameworks. In *COMMA*, 347–358.

Modgil, S., and Prakken, H. 2013. A General Account of Argumentation with Preferences. *Artif. Intell.* 195:361–397.

Modgil, S., and Prakken, H. 2014. The ASPIC+ Framework for Structured Argumentation: A Tutorial. *Argum.&Comput.* 5(1):31–62.

Modgil, S. 2009. Reasoning About Preferences in Argumentation Frameworks. *Artif. Intell.* 173(9-10):901–934.

Prakken, H., and Sartor, G. 1999. A System for Defeasible Argumentation, with Defeasible Priorities. In Wooldridge, M., and Veloso, M., eds., *Artif. Intell. Today*, volume 1600 of *Lecture Notes in Computer Science*. Springer. 365–379.

Rahwan, I., and Simari, G. R. 2009. *Argumentation in Artificial Intelligence*. Springer.

Rintanen, J. 1998. Complexity of Prioritized Default Logics. *J. Artif. Intell. Res.* 9:423–461.

Rott, H. 2001. *Change, Choice and Inference: A Study of Belief Revision and Nonmonotonic Reasoning*. Oxford University Press.

Sakama, C., and Inoue, K. 1996. Representing Priorities in Logic Programs. In *JICSLP*, 82–96.

Simari, G. R., and Loui, R. 1992. A Mathematical Treatment of Defeasible Reasoning and Its Implementation. *Artif. Intell.* 53(2-3):125–157.

Šimko, A. 2014. *Logic Programming With Preferences On Rules*. Ph.D. Dissertation, Comenius University in Bratislava.

Stolzenburg, F.; García, A. J.; Chesñevar, C.; and Simari, G. R. 2003. Computing Generalized Specificity. *J. Appl. Non-Classical Logics* 13:87–113.

Toni, F. 2014. A Tutorial on Assumption-Based Argumentation. *Argum.&Comput.* 5(1):89–117.

Wakaki, T. 2014. Assumption-Based Argumentation Equipped with Preferences. In *PRIMA*, 116–132.

Zhang, Y., and Foo, N. Y. 1997. Answer Sets for Prioritized Logic Programs. In *ILPS*, 69–83.

Equational properties of stratified least fixed points

Zoltan Esik
Dept. of Computer Science
University of Szeged
Hungary

Article published in: Proc. Logic, Language, Information, and Computation - 22nd International Workshop, WoLLIC 2015, Bloomington, IN, USA, July 20-23, LNCS 9160, Valeria de Paiva, Ruy J. G. B. de Queiroz, Lawrence S. Moss, Daniel Leivant and Anjolina Grisi de Oliveira, Eds., pp. 174188, Springer, 2015. `http://arxiv.org/abs/1410.8111`. Full version is under consideration for publication in a journal.

Studies on Brutal Contraction and Severe Withdrawal: Preliminary Report

Marco Garapa
Universidade da Madeira
CIMA - Centro de Investigação
em Matemática e Aplicações*
marco@uma.pt

Eduardo Fermé
Universidade da Madeira
NOVA Laboratory for Computer Science
and Informatics (NOVA LINCS)†
ferme@uma.pt

Maurício D. L. Reis
Universidade da Madeira
CIMA - Centro de Investigação
em Matemática e Aplicações*
m_reis@uma.pt

Abstract

In this paper we study the class of brutal base contractions that are based on a bounded ensconcement and also the class of severe withdrawals which are based on bounded epistemic entrenchment relations that are defined by means of bounded ensconcements (using the procedure proposed by Mary-Anne Williams). We present axiomatic characterizations for each one of those classes of functions and investigate the interrelation among them.

1 Introduction

The central goal underlying the research area of *logic of theory change* is the study of the changes which can occur in the belief state of a rational agent when he receives new information.

The most well known model of theory change was proposed by Alchourrón, Gärdenfors, and Makinson (1985) and is, nowadays, known as the AGM model. Assuming that the belief state of an agent is modelled by a *belief set* (i.e. a logically closed set of sentences), this framework essentially provides a definition for contractions — i.e. functions that receive a sentence (representing the new information received by the agent), and return a belief set which is a subset of the original one that does not contain the received sentence. In the mentioned paper, the class of *partial meet contractions* was introduced and axiomatically characterized. Subsequently several constructive models have been presented for the class of contraction functions proposed in the AGM framework (such as the *system of spheres-based contractions* (Grove 1988), *safe/kernel contractions* (Alchourrón and Makinson 1985; Hansson 1994), and the *epistemic entrenchment-based contractions* (Gärdenfors 1988; Gärdenfors and Makinson 1988)). Also several adaptations and variations of those constructive models have been presented and studied in the literature as it is the case, for example, of *severe withdrawals* (or mild contractions or Rott's contractions) (Rott 1991; Rott and Pagnucco 1999) which results of *simplifying* the definition of *epistemic entrenchment-based contractions*.

Although the AGM model has quickly acquired the status of standard model of theory change, several researchers (for an overview see (Fermé and Hansson 2011)) have pointed out its inadequateness in several contexts and proposed several extensions and generalizations to that framework. One of the most relevant of the proposed extensions of the AGM model of contraction is to use sets of sentences not (necessarily) closed under logical consequence — which are designated belief bases — rather than belief sets to represent belief states.

Hence, several of the existing models (of AGM contractions) were generalized to the case when belief states are represented by belief bases instead of belief sets. Among those we emphasize the *ensconcement-based contractions* and the *brutal base contractions* (of belief bases) proposed in (Williams 1995), which can be seen as adaptations to the case of belief bases of the *epistemic entrenchment-based contractions* and of the *severe withdrawals*, respectively. In fact, the definitions of those operations are both based on the concept of *ensconcement*, which is an adaptation of the concept of epistemic entrenchment relation to the case of belief bases. In the mentioned paper Mary-Anne Williams has also presented a method for constructing an epistemic entrenchment from an ensconcement relation.

In the present paper we will study the interrelation among *brutal base contractions* (of belief bases) and *severe withdrawals* (of belief sets). More precisely, we will devote special attention to the class of *brutal base contractions* which are based on bounded ensconcements — the so-called *bounded brutal base contractions* — and also to the class of the so-called *ensconcement-based severe withdrawals*, which is formed by the severe withdrawals that are based on an epistemic entrenchment relation defined from a bounded ensconcement using Mary-Anne William's method. We shall provide axiomatic characterizations to each one of those classes of functions and study the interrelation among them.

This paper is organized as follows: Firstly we provide the notation and background needed for the rest of the paper. After that we provide axiomatic characterizations for the classes of *bounded brutal base contractions* and of *ensconcement-based severe withdrawals*. Furthermore we show how to define a *bounded brutal base contraction* from an *ensconcement-based severe withdrawal* and vice-versa.

[*] Supported by FCT - Fundação para a Ciência e a Tecnologia through project UID/MAT/04674/2013 (CIMA).

[†] Supported by FCT MCTES and NOVA LINCS UID/CEC/04516/2013.

Finally, we briefly summarize the main contributions of the paper. In the appendix we provide proofs for the theorems. Proofs for all the remaining results are available at http://www.cee.uma.pt/ferme/GFR16-full.pdf.

2 Background

2.1 Formal preliminaries

We will assume a language $\mathcal{L}$ that is closed under truth-functional operations and a consequence operator Cn for $\mathcal{L}$. Cn satisfies the standard Tarskian properties, namely inclusion ($A \subseteq Cn(A)$), monotony (if $A \subseteq B$, then $Cn(A) \subseteq Cn(B)$), and iteration ($Cn(A) = Cn(Cn(A))$). It is supraclassical and compact, and satisfies deduction (if $\beta \in Cn(A \cup \{\alpha\})$, then $(\alpha \to \beta) \in Cn(A)$). $A \vdash \alpha$ will be used as an alternative notation for $\alpha \in Cn(A)$, $\vdash \alpha$ for $\alpha \in Cn(\emptyset)$ and $Cn(\alpha)$ for $Cn(\{\alpha\})$. Upper-case letters denote subsets of $\mathcal{L}$. Lower-case Greek letters denote elements of $\mathcal{L}$.

A well-ranked preorder on a set X is a preorder such that every nonempty subset of X has a minimal member, and similarly an inversely well-ranked preorder on a set X is a preorder such that every nonempty subset of X has a maximal member. A total preorder on X is bounded if and only if it is both well-ranked and inversely well-ranked.[1]

2.2 AGM

The AGM model of belief change was proposed by Alchourrón, Gärdenfors, and Makinson (1985) and acquired the status of standard model of belief change. In this model beliefs are represented by a set of sentences closed under logical consequence. In the AGM framework there are three operations to be considered, namely expansion, contraction and revision. Expansion, consists of adding new information (represented by sentences) in the original set preserving logical closure. Contraction, consists of eliminating sentences from a belief set, in such a way that the remaining set does not imply a specified sentence. Revision, consists in incorporating a sentence in the original set, but (eventually) eliminating some sentences in order to retain consistency of the revised set. AGM has been characterized in, at least five, different ways: *Postulates*, *partial meet functions*, *epistemic entrenchment*,*safe/kernel contraction* and *Grove' sphere-systems* (for an overview see (Fermé and Hansson 2011)).

One of the Postulates included in the axiomatic characterization of the contraction operator is recovery:

(Recovery) $K \subseteq (K - \alpha) + \alpha$

Recovery is based in the principle that "it is reasonable to require that we get all of the beliefs [...] back again after first contracting and then expanding with respect to the same belief" (Gärdenfors 1982). Nevertheless, the *recovery* postulate have been criticized by several authors (Fuhrmann 1991; Hansson 1991; Levi 1991; Niederée 1991) as a general principle that contractions should hold. Alternative contraction models were proposed in which the *recovery* postulate does not hold, for instance: Levi Contraction (Levi 1991), Severe Withdrawal (Rott 1991; Rott and Pagnucco 1999) and Semi-contraction (Fermé 1998).

2.3 Epistemic Entrenchment

Epistemic entrenchment was introduced in (Gärdenfors 1988; Gärdenfors and Makinson 1988) and relies on the idea that contractions on a belief set K should be based on an ordering of its sentences according to their epistemic entrenchment. When a belief set K is contracted it is prefered to give up beliefs with lower entrechment over others with a higher entrechment. Gärdenfors proposed the following set of axioms that an epistemic entrechment order $\leq$ related to a belief set K should satisfy:

(EE1) If $\alpha \leq \beta$ and $\beta \leq \gamma$, then $\alpha \leq \gamma$ (Transitivity)
(EE2) If $\alpha \vdash \beta$, then $\alpha \leq \beta$ (Dominance)
(EE3) $\alpha \leq (\alpha \wedge \beta)$ or $\beta \leq (\alpha \wedge \beta)$ (Conjunctiveness)
(EE4) If $K \not\vdash \perp$, then $\alpha \notin K$ if and only if $\alpha \leq \beta$ for all β (Minimality)
(EE5) If $\beta \leq \alpha$ for all β, then $\vdash \alpha$ (Maximality)

If $\leq$ is well-ranked and inversely well-ranked, then the epistemic entrenchment is well-ranked and inversely well-ranked, and therefore is a bounded epistemic entrenchment. The relation $\leq$ of epistemic entrenchment is independent of the change functions in the sense that it does not refer to any contraction or revision function. In addition to stating the axioms of entrenchment, Gärdenfors proposed the following entrenchment-based contraction functions:

$(G_{\leq})$ $\beta \in K{-}\alpha$ if and only if $\beta \in K$ and, either $\vdash \alpha$ or $\alpha < (\alpha \vee \beta)$

The crucial clause of $(G_{\leq})$ is $\alpha < (\alpha \vee \beta)$. This clause can be justified with reference to the recovery postulate (Gärdenfors and Makinson 1988).

Severe withdrawal: Rott (1991) proposed a more intuitive alternative definition, later called *Severe withdrawal* (or mild contraction or Rott's contraction) (Rott and Pagnucco 1999):

$(R_{\leq})$ $\beta \in K{-}\alpha$ if and only if $\beta \in K$ and, either $\vdash \alpha$ or $\alpha < \beta$

Arló-Costa and Levi (2006) have analyzed it in terms of minimal loss of informational value. It has been shown to satisfy the implausible postulate of expulsiveness. (If $\not\vdash \alpha$ and $\not\vdash \beta$, then either $\alpha \notin K \div \beta$ or $\beta \notin K \div \alpha$) (Hansson 1999b). Lindström and Rabinowicz (1991) abstained from recommending either a particularly expulsive contraction (severe withdrawal) or a particularly retentive one (AGM contraction). They argued that these extremes should be taken as "upper" and "lower" bounds and that any "reasonable" contraction function should be situated

[1] In (Williams 1994a) a preorder in these conditions is designated by *finite*, however we think it is more adequate to use the denomination *bounded*.

between them. This condition was called the Lindström's and Rabinowicz's interpolation thesis (Rott 1995). Severe withdrawal was axiomatized independently by Rott and Pagnucco (1999) and by Fermé and Rodriguez (1998). The following set of postulates characterize severe withdrawals (Rott and Pagnucco 1999):

($\div$1) $K \div \alpha = Cn(K \div \alpha)$
($\div$2) $K \div \alpha \subseteq K$
($\div$3) If $\alpha \notin K$ or $\vdash \alpha$, then $K \subseteq K \div \alpha$
($\div$4) If $\nvdash \alpha$, then $\alpha \notin K \div \alpha$
($\div$6) If $Cn(\alpha) = Cn(\beta)$, then $K \div \alpha = K \div \beta$
($\div 7a$) If $\nvdash \alpha$, then $K \div \alpha \subseteq K \div (\alpha \wedge \beta)$
($\div$8) If $\alpha \notin K \div (\alpha \wedge \beta)$, then $K \div (\alpha \wedge \beta) \subseteq K \div \alpha$

Severe withdrawal also satisfies the following postulates:

($\div$10) If $\nvdash \alpha$ and $\alpha \in K \div \beta$, then $K \div \alpha \subseteq K \div \beta$.
(Linearity) Either $K \div \alpha \subseteq K \div \beta$ or $K \div \beta \subseteq K \div \alpha$.
(Expulsiveness) If $\nvdash \alpha$ and $\nvdash \beta$, then either $\alpha \notin K \div \beta$ or $\beta \notin K \div \alpha$.

Rott and Pagnucco (1999) showed that an alternative axiomatization of severe withdrawals consists of the postulates ($\div$1) to ($\div$4) and ($\div$6) and:

($\div$9) If $\alpha \notin K \div \beta$, then $K \div \beta \subseteq K \div \alpha$.

2.4 Ensconcement

Williams (1992; 1995) defines an *ensconcement* relation on a belief base A as a transitive and connected relation $\preceq$ that satisfies the following three conditions:[2]

($\preceq$1) If $\beta \in A \setminus Cn(\emptyset)$, then $\{\alpha \in A : \beta \prec \alpha\} \nvdash \beta$
($\preceq$2) If $\nvdash \alpha$ and $\vdash \beta$, then $\alpha \prec \beta$, for all $\alpha, \beta \in A$
($\preceq$3) If $\vdash \alpha$ and $\vdash \beta$, then $\alpha \preceq \beta$, for all $\alpha, \beta \in A$

($\preceq$ 1) says that the formulae that are strictly more ensconced than α do not (even conjointly) imply α. Conditions ($\preceq$ 2) and ($\preceq$ 3) say that tautologies are the most ensconced formulae. If $\preceq$ is well-ranked/inversely well-ranked, then the ensconcement $(A, \preceq)$ is well-ranked/inversely well-ranked. If $\preceq$ is both well-ranked and inversely well-ranked then it is a bounded ensconcement.

Given an ensconcement relation, a cut operator for $\alpha \in Cn(A)$ is defined by:

$$cut_{\preceq}(\alpha) = \{\beta \in A : \{\gamma \in A : \beta \prec \gamma\} \nvdash \alpha\}.$$

A proper cut for $\alpha \in \mathcal{L}$ is defined by:

$$cut_{\prec}(\alpha) = \{\beta \in A : \{\gamma \in A : \beta \preceq \gamma\} \nvdash \alpha\}$$

Observation 1 (Williams 1994a)
If $\alpha \in A,\ cut_{\prec}(\alpha) = \{\beta \in A : \alpha \prec \beta\}$

The previous observation says that when α is an explicit belief, its proper cut is the subset of A such that its members are strictly more ensconced than α. Other properties of *proper cut* are:

Observation 2 Let $(A, \preceq)$ be a bounded ensconcement and $\alpha, \beta \in Cn(A)$, then:

(a) Let $\nvdash \beta$. If $cut_{\prec}(\alpha) \subseteq cut_{\prec}(\beta)$, then $cut_{\preceq}(\alpha) \subseteq cut_{\preceq}(\beta)$.
(b) If $\vdash \beta$ and $\nvdash \alpha$, then $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$.

Intuitively, an ensconcement is to belief bases as epistemic entrenchment is to belief sets. Williams explores this relation:

Definition 3 (Williams 1994b) Let $(A, \preceq)$ be an ensconcement. For $\alpha, \beta \in L$, define $\leq_{\preceq}$ to be given by: $\alpha \leq_{\preceq} \beta$ if and only if either:
i) $\alpha \notin Cn(A)$, or ii) $\alpha, \beta \in Cn(A)$ and $cut_{\preceq}(\beta) \subseteq cut_{\preceq}(\alpha)$.

Observation 4 (Williams 1994b) If $(A, \preceq)$ is an ensconcement, then $\leq_{\preceq}$ is an epistemic entrenchment related to $Cn(A)$.

Observation 5 (Williams 1994b) Given an ensconcement $(A, \preceq)$, $\preceq$ is well-ranked (inversely well-ranked, bounded) if and only if $\leq_{\preceq}$ is well-ranked (inversely well-ranked, bounded).

2.5 Brutal Contraction

Mary-Anne Williams (Williams 1994b) defines two operators for base contraction: The first one inspired in AGM contraction (ensconcement-based contraction) and the second one inspired in severe withdraw (brutal contraction). In this paper we will focus in the second one. Brutal contraction, as Mary-Anne Williams says, "retains as little as necessary of the theory base".

Definition 6 (Williams 1994b) Let A be a belief base. An operation $-$ is a brutal base contraction on A if and only if there is an ensconcement relation $\preceq$ on A such that:

$\beta \in A - \alpha$ if and only if $\beta \in A$ and either (i) $\alpha \in Cn(\emptyset)$ or (ii) $\beta \in cut_{\prec}(\alpha)$

In (Garapa, Fermé, and Reis 2016) the following axiomatic characterization for brutal base contractions was presented:

Observation 7 (Garapa, Fermé, and Reis 2016) Let A be a belief base. An operator $-$ of A is a brutal base contraction on A if and only if it satisfies:

(Success) If $\nvdash \alpha$, then $A - \alpha \nvdash \alpha$
(Inclusion) $A - \alpha \subseteq A$
(Vacuity) If $A \nvdash \alpha$, then $A \subseteq A - \alpha$
(Failure) If $\vdash \alpha$, then $A - \alpha = A$
(Relative Closure) $A \cap Cn(A - \alpha) \subseteq A - \alpha$
(Strong Inclusion) If $A - \beta \nvdash \alpha$, then $A - \beta \subseteq A - \alpha$

[2] $\alpha \prec \beta$ means $\alpha \preceq \beta$ and $\beta \npreceq \alpha$. $\alpha =_{\preceq} \beta$ means $\alpha \preceq \beta$ and $\beta \preceq \alpha$.

(Uniform Behaviour) If $\beta \in A$, $A \vdash \alpha$ and $A - \alpha = A - \beta$, then $\alpha \in Cn(A - \beta \cup \{\gamma \in A : A - \beta = A - \gamma\})$

The following observation lists some other well-known postulates which are satisfied by the brutal base contraction functions.

Observation 8 (Garapa, Fermé, and Reis 2016) Let A be a belief base and $-$ an operator on A that satisfies *success, inclusion, vacuity, failure, relative closure, strong inclusion* and *uniform behaviour*. Then $-$ satisfies:

(a) If $\alpha \in A \setminus A - \beta$, then $A - \beta \subseteq A - \alpha$.

(b) If $A - \alpha \subset A - \beta$, then $A - \beta \vdash \alpha$.

(c) If $\vdash \alpha$ and $\alpha \in A$, then $\alpha \in A - \beta$.

(d) If $\vdash \alpha \leftrightarrow \beta$, then $A - \alpha = A - \beta$. (Extensionality)

3 Bounded Brutal Base Contraction Functions

In this subsection we introduce the bounded brutal base contractions and obtain an axiomatic characterization for that class of functions.

Definition 9 Let A be a belief base. An operation $-$ is a bounded brutal base contraction on A if and only if it is a brutal base contraction based on a bounded ensconcement.

We introduce the following postulates:

(Upper Bound) For every non-empty set $X \subseteq A$ of nontautological formulae, there exists $\alpha \in X$ such that $A - \beta \subseteq A - \alpha$ for all $\beta \in X$
(Lower Bound) For every non-empty set $X \subseteq A$ of nontautological formulae, there exists $\alpha \in X$ such that $A - \alpha \subseteq A - \beta$ for all $\beta \in X$
(Clustering) If $\beta \in A$, then there exists $\alpha \in A \cup Cn(\emptyset)$ such that $A - \alpha = A - \beta \cup \{\gamma \in A : A - \beta = A - \gamma\}$

Upper Bound (respectively Lower Bound) states that every non-empty set of nontautological formulae of A contains an element which is such that the result of contracting A by that sentence is a superset (respectively a subset) of any set which results of contracting A by one of the remaining sentences of that subset.

Clustering asserts that for any sentence β in A there exists some sentence α in $A \cup Cn(\emptyset)$ such that the result of the contraction of α from A is the set consisting of the union of the result of contracting A by β with the set formed by all the sentences of A which are such that the result of contracting it from A coincides with the result of contracting A by β.

The two following observations present some interrelations among the above proposed postulates and some of the of the postulates included in the axiomatic characterization that was obtained for the class of brutal base contraction.

Observation 10 Let A be a belief base and $-$ an operator on A that satisfies *success, inclusion, failure, relative closure, strong inclusion* and *lower bound*. Then $-$ satisfies *clustering*.

Observation 11 Let A be a belief base and $-$ an operator on A that satisfies *failure, success, strong inclusion and clustering*. Then $-$ satisfies *uniform behaviour*.

We are now in a position to present an axiomatic characterization for the class of bounded brutal base contractions.

Theorem 12 (Axiomatic characterization of bounded brutal base contraction functions) Let A be a belief base. An operator $-$ on A is a bounded brutal base contraction on A if and only if it satisfies *success, inclusion, vacuity, failure, relative closure, lower bound, upper bound* and *strong inclusion*.

The following observation exposes another relevant property of the bounded brutal base contractions which will be useful further ahead. More precisely, it asserts that for any non-tautological sentence α which is deducible from A it holds that the result of contracting A by α coincides with the result of the contraction of A by some sentence explicitly included in A.

Observation 13 Let A be a belief base and $-$ an operator on A that satisfies *success, inclusion, failure, relative closure, strong inclusion and* lower bound. Then $-$ satisfies:
For all $\alpha \in Cn(A) \setminus Cn(\emptyset)$ there exists $\beta \in A$ such that $A - \alpha = A - \beta$.

4 Relation between Bounded Brutal Base Contraction and Ensconcement-based Severe Withdrawal

In this section we will define and axiomatically characterize a particular kind of severe withdrawals which we will show to be the contraction functions that correspond to the bounded brutal base contractions in the context of belief set contractions.

We start by noticing that, given a bounded ensconcement $(A, \preceq)$, we can combine Definitions 3 and $(R_{\leq})$ in order to define a contraction function on the belief set $Cn(A)$. This kind of functions is formally introduced in the following definition.

Definition 14 $\div$ is an ensconcement-based withdrawal related to $(A, \preceq)$ if and only if $(A, \preceq)$ is a bounded ensconcement such that $Cn(A) \div \alpha = Cn(A) \div_{\leq_{\preceq}} \alpha$, where $\leq_{\preceq}$ is the epistemic entrenchment with respect to $Cn(A)$ defined by Definition 3 and $\div_{\leq_{\preceq}}$ is the severe withdrawal on $Cn(A)$ defined by $(R_{\leq})$.

Comparing the above definition with Definitions 6 and 9 it becomes clear that there is a strong interrelation among the ensconcement-based severe withdrawals and the (bounded) brutal base contractions. That interrelation is explicitly presented in the two following theorems. More precisely, given a bounded ensconcement $(A, \preceq)$, these two results expose how the $\preceq$-based brutal contraction on A can be defined from the ensconcement-based withdrawal related to $(A, \preceq)$ and, vice-versa, how the latter can be defined by means of the former.

Theorem 15 Let $(A, \preceq)$ be a bounded ensconcement, $-$ be the $\preceq$-based brutal contraction, and $\div_{\leq_{\preceq}}$ be the ensconcement-based severe withdrawal related to $(A, \preceq)$, then $A - \alpha = (Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A$.

Theorem 16 Let $(A, \preceq)$ be a bounded ensconcement, $-$ be the $\preceq$-based brutal contraction, and $\div_{\leq_{\preceq}}$ be the ensconcement-based severe withdrawal related to $(A, \preceq)$, then $Cn(A) \div_{\leq_{\preceq}} \alpha = Cn(A - \alpha)$.

4.1 Axiomatic Characterization of ensconcement-based severe withdrawals

In this subsection we will present an axiomatic characterization for the class of ensconcement-based severe withdrawals. To do that we must start by introducing the following postulate:
(Base-reduction) If $Cn(A) \div \alpha \vdash \beta$, then $(Cn(A) \div \alpha) \cap A \vdash \beta$

This postulate essentially states that that the result of contracting the belief set $Cn(A)$ by any sentence α coincides with the logical closure of some subset of A. Indeed, it is not hard to see that base-reduction is equivalent to the following postulate: $\forall \alpha \exists A' \subseteq A : Cn(A') = Cn(A) \div \alpha$ (which is very similar to the postulate of *finitude* proposed by Hansson (1999a)).

The following observation highlights that for a severe withdrawal that satisfies the postulates of *base-reduction* and *lower bound* it also holds that for any non-tautological sentence α in $Cn(A)$ the result of the contraction of $Cn(A)$ by α coincides with the result of the contraction of $Cn(A)$ by some sentence in A.

Observation 17 Let $\div$ be an operator on $Cn(A)$ that satisfies $(\div 1)$, $(\div 2)$, $(\div 4)$, $(\div 9)$, *base-reduction* and *lower bound*, then for all $\alpha \in Cn(A) \setminus Cn(\emptyset)$ there exists $\beta \in A$ such that $Cn(A) \div \alpha = Cn(A) \div \beta$.

We are now in a position to present the following axiomatic characterization for the ensconcement-based severe withdrawals.

Theorem 18 Let A be a belief base and $\div$ be an operator on $Cn(A)$. $\div$ satisfies $(\div 1)$ to $(\div 4)$, $(\div 6)$, $(\div 9)$, *base-reduction, upper bound* and *lower bound* if and only if there exists a bounded ensconcement such that $\div$ is an ensconcement-based withdrawal related to $(A, \preceq)$.

Theorems 15 and 16 expose how a base contraction function can be defined from a belief set contraction function and, vice-versa. Combining those two results with the axiomatic characterizations presented in Theorems 12 and 18 we can obtain the following results which highlight the correspondence among sets of postulates for base contraction and sets of postulates for belief set contraction.

Corollary 19 *An operator $-$ on A satisfies* success, inclusion, vacuity, failure, relative closure, strong inclusion, upper bound *and* lower bound *if and only if there exists an operator $\div$ on $Cn(A)$ that satisfies $(\div 1)$ to $(\div 4)$, $(\div 6)$, $(\div 9)$,* base-reduction, upper bound *and* lower bound *such that: $A - \alpha = Cn(A \div \alpha) \cap A$.*

Corollary 20 *An operator $\div$ on $Cn(A)$ satisfies $(\div 1)$ to $(\div 4)$, $(\div 6)$, $(\div 9)$,* base-reduction, upper bound *and* lower bound *if and only if there exists an operator $-$ on A that satisfies* success, inclusion, vacuity, failure, relative closure, strong inclusion, upper bound *and* lower bound *such that: $Cn(A) \div \alpha = Cn(A - \alpha)$.*

The two following observations consist of a slight refinement of the right to left part of Corollary 20. More precisely these results specify more precisely which properties of the belief base contraction are needed in order to assure that the belief set contraction obtained from it as exposed in Theorem 16 satisfies certain postulates.

Observation 21 Let A be a belief base and $-$ be an operator on A that satisfies *success, inclusion, vacuity, failure, relative closure* and *strong inclusion*. If $\div$ is an operator on $Cn(A)$ defined by $Cn(A) \div \alpha = Cn(A - \alpha)$ then $\div$ satisfies $(\div 1)$ to $(\div 4)$, $(\div 6)$, $(\div 9)$ and *base-reduction*.

Observation 22 Let A be a belief base and $-$ be an operator on A that satisfies *success, inclusion, failure, relative closure, upper bound, lower bound* and *strong inclusion*. If $\div$ is an operator on $Cn(A)$ defined by $Cn(A) \div \alpha = Cn(A - \alpha)$ then $\div$ satisfies *upper bound* and *lower bound*.

5 Conclusions

We have presented an axiomatic characterizations for the subclass of brutal base contractions formed by the brutal contractions that are based on a bounded ensconcement relation. We have also introduced and axiomatically characterized the class of ensconcement-based severe withdrawals which is formed by the severe withdrawals that are based on epistemic entrenchment relations which are obtained from an ensconcement relation using the construction proposed by Mary-Anne Williams. Some results were presented concerning the interrelation among the classes of bounded brutal base contractions and of ensconcement-based severe withdrawals. Finally we presented some results relating base contraction postulates and belief set contraction postulates by means of explicit definitions of belief set contractions from base contractions and vice-versa.

Acknowledgements

We wish to thank the three reviewers for their comments which have contributed to the improvement of this paper.

Appendix: Proofs

Previous Lemmas

Lemma 23 (Fermé, Krevneris, and Reis 2008)

(a) If $\not\vdash \alpha, cut_{\prec}(\alpha) \not\vdash \alpha$.

(b) If $A \not\vdash \alpha, cut_{\prec}(\alpha) = A$.

(c) If $\beta \vdash \alpha$, then $cut_{\prec}(\alpha) \subseteq cut_{\prec}(\beta)$.

(d) If $\alpha \preceq \beta$, then $cut_{\prec}(\beta) \subseteq cut_{\prec}(\alpha)$.

(e) If $cut_{\prec}(\alpha) \vdash \beta$, then $cut_{\prec}(\alpha \wedge \beta) = cut_{\prec}(\alpha)$.

(f) If $cut_{\prec}(\alpha) \not\vdash \beta$, then $cut_{\prec}(\alpha \wedge \beta) = cut_{\prec}(\beta)$.

Lemma 24 (Rott and Pagnucco 1999, Observation 19(ii)) If $\div$ is a severe withdrawal function, then $\div$ can be represented as an entrenchement-based withdrawal where the relation $\leq$ on which $\div$ is based is obtained by
(Def $\leq$ from $\div$) $\alpha \leq \beta$ if and only if $\alpha \notin K \div \beta$ or $\vdash \beta$
and $\leq$ satisfies (EE1) to (EE5).

Lemma 25 Let $(A, \preceq)$ be a bounded ensconcement and $cut_{\preceq}(\alpha) \neq \emptyset$. Then there exists $\beta \in cut_{\preceq}(\alpha)$ such that $cut_{\preceq}(\beta) = cut_{\preceq}(\alpha)$.

Lemma 26 Let $(A, \preceq)$ be a bounded ensconcement and $\alpha \in Cn(A)$. Then $cut_{\preceq}(\alpha) \vdash \alpha$.

Lemma 27 Let $(A, \preceq)$ be a bounded ensconcement and $\alpha, \beta \in Cn(A)$. If $cut_{\prec}(\alpha) \subset cut_{\prec}(\beta)$, then $cut_{\preceq}(\alpha) \subset cut_{\preceq}(\beta)$.

Proofs

Proof of Theorem 12

From bounded brutal base contraction to postulates

Let $-$ be a bounded brutal base contraction operator on A. By Observation 7 $-$ satisfies *success, inclusion, vacuity, failure, relative closure* and *strong inclusion*. It remains to show that $-$ satisfies *upper bound* and *lower bound*.

Upper Bound Let $X \subseteq A$ be a non empty set of non-tautological formulae. Since $\preceq$ is well ranked there exists $\beta \in X$ such that $\beta \preceq \alpha$ for all $\alpha \in X$. Hence, by Lemma 23 (d), there exists $\beta \in X$ for all $\alpha \in X$ such that $cut_{\prec}(\alpha) \subseteq cut_{\prec}(\beta)$. Therefore, by definition of $-$ there exists $\beta \in X$ for all $\alpha \in X$ such that $A - \alpha \subseteq A - \beta$.

Lower Bound Analogous to *upper bound*.

From postulates to bounded brutal base contraction

Let $-$ be an operator on A that satisfies *success, inclusion, vacuity, failure, relative closure, lower bound, upper bound* and *strong inclusion*. From Observation 10 and Observation 11 it follows that $-$ satisfies *uniform behaviour*. Let $\preceq$ be defined by:

$$\alpha \preceq \beta \text{ iff } \begin{cases} A - \beta \subseteq A - \alpha \text{ and} \nvdash \alpha \\ \text{or} \\ \vdash \beta \end{cases}$$

According to the Postulates to Construction part of the proof of Observation 7 $\preceq$ satisfies ($\preceq 1$) **-** ($\preceq 3$) and is such that

$$A - \alpha = \begin{cases} cut_{\prec}(\alpha) & \text{if} \nvdash \alpha \\ A & \text{otherwise} \end{cases}$$

It remains to prove that $\preceq$ is bounded. To do so we must prove that $\preceq$ is well-ranked and inversely well-ranked.

($\preceq$ is well-ranked) Let $X \neq \emptyset$ and $X \subseteq A$. We will prove by cases:

Case 1) All formulae in X are tautologies. Let β be one of those formulas. Hence by ($\preceq 3$) $\beta \preceq \alpha$ for all $\alpha \in X$.

Case 2) All formulae in X are non-tautological. By *upper bound* there exists $\beta \in X$ such that $A - \alpha \subseteq A - \beta$ for all $\alpha \in X$. Hence, by definition of $\preceq$, there exists $\beta \in X$ such that $\beta \preceq \alpha$ for all $\alpha \in X$.

Case 3) There are some formulae in X, that are tautological and others that are not. Consider $X' = X \setminus Cn(\emptyset)$. Hence, by the previous case, there exists $\beta \in X'$ such that $\beta \preceq \alpha'$ for all $\alpha' \in X'$. Therefore, it follows from ($\preceq 3$) that $\beta \preceq \alpha$ for all $\alpha \in X$.

($\preceq$ is inversely well-ranked) Let $X \neq \emptyset$ and $X \subseteq A$. We will prove by cases:

Case 1) There are some $\beta \in X$ such that $\vdash \beta$. Then, by definition of $\preceq$, $\alpha \preceq \beta$ for all $\alpha \in X$.

Case 2) All formulae in X are non-tautological. By *lower bound* there exists $\beta \in X$ such that $A - \beta \subseteq A - \alpha$ for all $\alpha \in X$. Hence, by definition of $\preceq$, there exists $\beta \in X$ such that $\alpha \preceq \beta$ for all $\alpha \in X$. ■

Proof of Theorem 15

We will prove by cases:

Case 1) $\vdash \alpha$. It follows that $A - \alpha = A$ and $(Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A = A$.

Case 2) $A \nvdash \alpha$. It follows that $(Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A = A$ and that $A - \alpha = cut_{\prec}(\alpha)$. By Lemma 23 (b), it follows that $cut_{\prec}(\alpha) = A$.

Case 3)$A \vdash \alpha$ and $\nvdash \alpha$.

We will prove that $A - \alpha = (Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A$ by double inclusion.

($\subseteq$) Let $\beta \in A - \alpha$. It follows that $\beta \in A$. It remains to prove that $\beta \in Cn(A) \div_{\leq_{\preceq}} \alpha$, i.e. that $\beta \in \{\psi \in Cn(A) : cut_{\preceq}(\psi) \subset cut_{\preceq}(\alpha)\}$.

If $\vdash \beta$. It follows trivially by Observation 2 (b).

Assume now that $\nvdash \beta$. $\beta \in cut_{\prec}(\alpha)$. Hence $cut_{\prec}(\beta) \subset cut_{\prec}(\alpha)$. It follows, from Lemma 27 that $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$.

($\supseteq$) Let $\beta \in (Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A$. If $\vdash \beta$, then it follows from ($\preceq 2$) that $\{\psi \in A : \beta \preceq \psi\} \subseteq Cn(\emptyset)$. Therefore, since $\nvdash \alpha$, it follows that $\beta \in cut_{\prec}(\alpha) = A - \alpha$. Assume now that $\nvdash \beta$. From $\beta \in (Cn(A) \div_{\leq_{\preceq}} \alpha) \cap A$ it follows that $\beta \in A$ and $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$. Hence there exists $\gamma \in A$ such that $\gamma \in cut_{\preceq}(\alpha)$ and $\gamma \notin cut_{\preceq}(\beta)$. Hence, $\{\psi \in A : \gamma \prec \psi\} \nvdash \alpha$ and $\{\psi \in A : \gamma \prec \psi\} \vdash \beta$. Assume by *reductio* that $\beta \notin A - \alpha$ i.e. that $\beta \notin cut_{\prec}(\alpha)$. Hence, $\{\psi \in A : \beta \preceq \psi\} \vdash \alpha$. From $\{\psi \in A : \beta \preceq \psi\} \vdash \alpha$ and $\{\psi \in A : \gamma \prec \psi\} \nvdash \alpha$ it follows that $\beta \preceq \gamma$. Therefore, since $\{\psi \in A : \gamma \prec \psi\} \vdash \beta$, it follows that $\{\psi \in A : \beta \prec \psi\} \vdash \beta$ which contradicts ($\preceq 1$). ■

Proof of Theorem 16

We will prove by cases:

Case 1) $\vdash \alpha$. Then $Cn(A) \div_{\leq_{\preceq}} \alpha = Cn(A)$ and $A - \alpha = A$. Hence $Cn(A - \alpha) = Cn(A) = Cn(A) \div_{\leq_{\preceq}} \alpha$.

Case 2) $A \nvdash \alpha$. Then $Cn(A) \div_{\leq_{\preceq}} \alpha = Cn(A)$ and, by Lemma 23 (b), $A - \alpha = cut_{\prec}(\alpha) = A$. Hence $Cn(A - \alpha) = Cn(A) = Cn(A) \div_{\leq_{\preceq}} \alpha$.

Case 3)$A \vdash \alpha$ and $\nvdash \alpha$. Hence $Cn(A) \div_{\preceq} \alpha = \{\psi \in Cn(A) : \alpha <_{\preceq} \psi\} = \{\psi \in Cn(A) : cut_{\preceq}(\psi) \subset cut_{\preceq}(\alpha)\}$. We will prove that $Cn(A - \alpha) = Cn(A) \div_{\leq_{\preceq}} \alpha$ by double inclusion.

($\subseteq$) Let $\beta \in Cn(A - \alpha)$. If $\vdash \beta$, then $\beta \in Cn(A)$ and, by Observation 2 (b), $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$. Hence $\beta \in Cn(A) \div_{\preceq} \alpha$.

Assume now that $\nvdash \beta$. From $\beta \in Cn(A - \alpha)$ it follows that $cut_{\prec}(\alpha) \vdash \beta$. Hence, by Lemma 23 (e), $cut_{\prec}(\alpha \wedge \beta) = cut_{\prec}(\alpha)$. From $\alpha \wedge \beta \vdash \beta$ by Lemma 23 (c) it follows that $cut_{\prec}(\beta) \subseteq cut_{\prec}(\alpha \wedge \beta)$. Hence $cut_{\prec}(\beta) \subseteq cut_{\prec}(\alpha)$. From which, together with Lemma 23 (a) and $cut_{\prec}(\alpha) \vdash \beta$ it follows that $cut_{\prec}(\beta) \subset cut_{\prec}(\alpha)$. Hence, by Lemma 27, it follows that $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$. Therefore, since $\beta \in Cn(A)$, it follows that $\beta \in Cn(A) \div_{\leq_{\preceq}} \alpha$.

($\supseteq$) Let $\beta \in Cn(A) \div_{\leq_{\preceq}} \alpha$. Hence, $\beta \in Cn(A)$ and $cut_{\preceq}(\beta) \subset cut_{\preceq}(\alpha)$. Assume by *reductio* that $\beta \notin Cn(A -$

α). Therefore $cut_{\prec}(\alpha) \not\vdash \beta$. By Lemma 23 (f) it follows that $cut_{\prec}(\alpha \wedge \beta) = cut_{\prec}(\beta)$. From $\alpha \wedge \beta \vdash \alpha$, by Lemma 23 (c), it follows that $cut_{\prec}(\alpha) \subseteq cut_{\prec}(\beta)$. From Observation 2 (a) it follows that $cut_{\preceq}(\alpha) \subseteq cut_{\preceq}(\beta)$. Contradiction. ■

Proof of Theorem 18

($\Leftarrow$) Let $\div$ be an ensconcement-based withdrawal related to $(A, \preceq)$ and let $\leq = \leq_{\preceq}$. Hence $\div$ satisfies the postulates for severe withdrawals. It remains to show that $\div$ satisfies: *base-reduction, upper bound* and *lower bound*.

Upper Bound: Let $\div$ be an ensconcement-based withdrawal related to $(A, \preceq)$. Let $X \neq \emptyset$ and $X \subseteq Cn(A) \setminus Cn(\emptyset)$. From Observation 5, since $(A, \preceq)$ is a bounded ensconcement, it follows that $\leq_{\preceq}$ is bounded. Hence, there exists $\beta \in X$ such that $\beta \leq \alpha$ for all $\alpha \in X$. We will prove that $Cn(A) \div \alpha \subseteq Cn(A) \div \beta$ for all $\alpha \in X$. Let $\gamma \in Cn(A) \div \alpha$. Hence, by definition of $\div$, $\gamma \in Cn(A)$ and $\alpha < \gamma$. By EE1, since $\beta \leq \alpha$ and $\alpha < \gamma$ it follows that $\beta < \gamma$. Hence $\gamma \in Cn(A) \div \beta$. Therefore $Cn(A) \div \alpha \subseteq Cn(A) \div \beta$.

Lower Bound: Analogous to *upper bound*.

Base-reduction: Let $Cn(A) \div \alpha \vdash \beta$. We will prove that $(Cn(A) \div \alpha) \cap A \vdash \beta$ by cases:

Case 1) $\vdash \beta$. Follows trivially.

Case 2) $\alpha \notin Cn(A)$ or $\vdash \alpha$. Follows trivially by $(R_{\leq})$.

Case 3) $\not\vdash \beta, \alpha \in Cn(A)$ and $\not\vdash \alpha$. From $Cn(A) \div \alpha \vdash \beta$ it follows, by $(R_{\leq})$, that $X \vdash \beta$ where $X = \{\psi \in Cn(A) : cut_{\preceq}(\psi) \subset cut_{\preceq}(\alpha)\}$. $X \setminus Cn(\emptyset) \neq \emptyset$, since $\not\vdash \beta$. Let $\psi \in X \setminus Cn(\emptyset)$. Assume that $cut_{\preceq}(\psi) = \emptyset$ and let $\theta \in Cn(\emptyset)$. Hence, by EE5, it follows that $\psi < \theta$. Hence, by Definition 3, $cut_{\preceq}(\theta) \subset cut_{\preceq}(\psi) = \emptyset$. Contradiction. Hence $cut_{\preceq}(\psi) \neq \emptyset$. From Lemma 25, and since $\preceq$ is bounded, it follows that there exists $\delta \in cut_{\preceq}(\psi)$ such that $cut_{\preceq}(\delta) = cut_{\preceq}(\psi)$. Let $Y = \{\mu \in A : cut_{\preceq}(\mu) \subset cut_{\preceq}(\alpha)\}$. Let $\mu_1 \in Y$ such that $\mu_1 \preceq \mu$ for all $\mu \in Y$. Let $\lambda \in cut_{\preceq}(\mu_1)$. Hence $cut_{\preceq}(\lambda) \subseteq cut_{\preceq}(\mu_1)$, from which follows that $cut_{\preceq}(\lambda) \subset cut_{\preceq}(\alpha)$. Therefore $\lambda \in Y$. Let $\phi \in Y$. It follows that $\mu_1 \preceq \phi$. Hence $\phi \in cut_{\preceq}(\mu_1)$. Therefore $Y = cut_{\preceq}(\mu_1)$. By Lemma 26 $cut_{\preceq}(\psi) \vdash \psi$. Hence, since $cut_{\preceq}(\delta) = cut_{\preceq}(\psi)$ it follows that $cut_{\preceq}(\delta) \vdash \psi$. From $cut_{\preceq}(\delta) \subset cut_{\preceq}(\alpha)$ it follows that $\delta \in Y$. Hence $\mu_1 \preceq \delta$. Therefore $cut_{\preceq}(\delta) \subseteq cut_{\preceq}(\mu_1) = Y$, and so $Y \vdash \psi$. Hence, for all $\psi \in Cn(A) \div \alpha$ it follows that $Y \vdash \psi$. Therefore, since $Cn(A) \div \alpha \vdash \beta$, it follows that $Y \vdash \beta$. $Y \subseteq (Cn(A) \div \alpha) \cap A$. Hence $(Cn(A) \div \alpha) \cap A \vdash \beta$.

($\Rightarrow$) Let A be a belief base and $\div$ be an operator on $Cn(A)$. $\div$ satisfies $(\div 1)$ to $(\div 4)$, $(\div 6)$, $(\div 9)$, *base-reduction, upper bound* and *lower bound*. Let $\preceq$ be a binary relation on A defined by:

$\alpha \preceq \beta$ if and only if $\alpha \notin Cn(A) \div \beta$ or $\vdash \beta$.

We will prove that $\preceq$ is a bounded ensconcement.

($\preceq$1) Let $\gamma \in A \setminus Cn(\emptyset)$ we must show that $H = \{\alpha \in A : \gamma \prec \alpha\} \not\vdash \gamma$. It is enough to show that $H \setminus Cn(\emptyset) \not\vdash \gamma$. Let $\alpha \in A \setminus Cn(\emptyset)$ and $\gamma \prec \alpha$. Then, $\gamma \preceq \alpha$ and $\alpha \not\preceq \gamma$. Hence, by definition of $\preceq$, it follows that $\gamma \notin Cn(A) \div \alpha$, $\alpha \in Cn(A) \div \gamma$ and $\not\vdash \gamma$. $H \subseteq Cn(A) \div \gamma$ where, $\not\vdash \gamma$. Hence, since by $(\div 4)$ $Cn(A) \div \gamma \not\vdash \gamma$ it follows that $H \not\vdash \gamma$.

($\preceq$2) Let $\alpha, \beta \in A$ be such that $\not\vdash \alpha$ and $\vdash \beta$. From $\vdash \beta$ it follows, by definition of $\preceq$, that $\alpha \preceq \beta$. Assume by *reductio* that $\not\vdash \alpha$, $\vdash \beta$ and $\beta \preceq \alpha$. Hence, by definition of $\preceq$, $\beta \notin Cn(A) \div \alpha$ or $\vdash \alpha$. Contradiction, since $\not\vdash \alpha$ and by $(\div 1)$ $\beta \in Cn(A) \div \alpha$.

($\preceq$3) Follows trivially by definition of $\preceq$.

($\preceq$ is transitive) Let $\alpha \preceq \beta$ and $\beta \preceq \gamma$. Hence, by definition of $\preceq$, it follows that ($\alpha \notin Cn(A) \div \beta$ or $\vdash \beta$) and ($\beta \notin Cn(A) \div \gamma$ or $\vdash \gamma$). Hence, $\alpha \notin Cn(A) \div \beta$ and ($\beta \notin Cn(A) \div \gamma$ or $\vdash \gamma$) or ($\vdash \beta$ and ($\beta \notin Cn(A) \div \gamma$ or $\vdash \gamma$)). Hence, we have four cases to consider:

Case 1) $\alpha \notin Cn(A) \div \beta$ and $\beta \notin Cn(A) \div \gamma$. From $(\div 9)$ it follows that $Cn(A) \div \gamma \subseteq Cn(A) \div \beta$. Hence, $\alpha \notin Cn(A) \div \gamma$. Therefore $\alpha \preceq \gamma$, by definition of $\preceq$.

Case 2) $\alpha \notin Cn(A) \div \beta$ and $\vdash \gamma$. $\alpha \preceq \gamma$ follows trivially by definition of $\preceq$.

Case 3) $\vdash \beta$ and $\beta \notin Cn(A) \div \gamma$. Contradicts $(\div 1)$.

Case 4) $\vdash \beta$ and $\vdash \gamma$. $\alpha \preceq \gamma$ follows trivially by definition of $\preceq$.

($\preceq$ is connected) Let $\alpha \not\preceq \beta$. Hence $\alpha \in Cn(A) \div \beta$ and $\not\vdash \beta$. We will consider two cases:

Case 1) $\vdash \alpha$. Hence $\beta \preceq \alpha$, by definition of $\preceq$.

Case 2) $\not\vdash \alpha$. Hence, by $\div$ *expulsiveness*, $\beta \notin Cn(A) \div \alpha$. Therefore, by definition of $\preceq$, $\beta \preceq \alpha$.

($\preceq$ is well-ranked) Let $X \subseteq A$ a non empty set. We will prove by cases:

Case 1) $X \subseteq Cn(\emptyset)$. Trivial.

Case 2) $X \not\subseteq Cn(\emptyset)$. Let $X' = X \setminus Cn(\emptyset)$. Hence, by $\div$ *upper bound* there exists $\beta \in X'$ such that $Cn(A) \div \alpha \subseteq Cn(A) \div \beta$ for all $\alpha \in X'$. By $(\div 4)$ $\beta \notin Cn(A) \div \alpha$ for all $\alpha \in X'$. Hence, by definition of $\preceq$, there exists $\beta \in X'$ such that $\beta \preceq \alpha$ for all $\alpha \in X'$. If $X = X'$ trivial. Assume now that $X \neq X'$. Let $\gamma \in X \setminus X'$. Hence $\vdash \gamma$ and by ($\preceq$ 2) it follows that $\beta \preceq \gamma$. Therefore, there exists $\beta \in X$ such that $\beta \preceq \alpha$ for all $\alpha \in X$.

($\preceq$ is inversely well-ranked) Let $X \subseteq A$ a non empty set. We will consider two cases:

Case 1)$X \cap Cn(\emptyset) \neq \emptyset$. Let $\beta \in X \cap Cn(\emptyset)$ hence, by definition of $\preceq$, $\alpha \preceq \beta$ for all $\alpha \in X$.

Case 2)$X \cap Cn(\emptyset) = \emptyset$. Hence, by $\div$ *lower bound*, there exists $\beta \in X$ such that $Cn(A) \div \beta \subseteq Cn(A) \div \alpha$, for all $\alpha \in X$. By $(\div 4)$ $\alpha \notin Cn(A) \div \beta$, for all $\alpha \in X$. Hence, by definition of $\preceq$ there exists $\beta \in X$ such that $\alpha \preceq \beta$, for all $\alpha \in X$.

We have proved that $\preceq$ is a bounded ensconcement. Let $\leq_{\preceq}$ be as in Definition 3. According to Observation 4 and Observation 5 $\leq_{\preceq}$ is a bounded epistemic entrenchment related to $Cn(A)$. It remains to show that $Cn(A) \div \alpha = Cn(A) \div_{\leq_{\preceq}} \alpha$, where $\div_{\leq_{\preceq}}$ is defined (as in $(R_{\leq})$)by:

$$Cn(A) \div_{\leq_{\preceq}} \alpha =$$
$$\begin{cases} Cn(A) \cap \{\psi : \alpha <_{\preceq} \psi\} & \text{if } \alpha \in Cn(A) \text{and} \not\vdash \alpha \\ Cn(A) & \text{otherwise} \end{cases}$$

According to Lemma 24 and since $\div$ is a severe withdrawal function, the epistemic entrenchment $\leq$ on which $\div$ is based on is such that: $\alpha \leq \beta$ if and only if $\alpha \notin Cn(A) \div \beta$ or $\vdash \beta$. Thus to prove that $Cn(A) \div \alpha = Cn(A) \div_{\leq_{\preceq}} \alpha$ it is enough to show that:

$\alpha \leq_{\preceq} \beta$ if and only if $\alpha \notin Cn(A) \div \beta$ or $\vdash \beta$.

(⇒) Let $\alpha \leq_{\preceq} \beta$. Hence, by definition of $\leq_{\preceq}$, $\alpha \leq_{\preceq} \beta$ if and only if:
i) $\alpha \notin Cn(A)$, or
ii) $\alpha, \beta \in Cn(A)$ and $\mathit{cut}_{\preceq}(\beta) \subseteq \mathit{cut}_{\preceq}(\alpha)$.
We will prove by cases:
Case 1) $\alpha \notin Cn(A)$. Then, by $(\div 2)$, $\alpha \notin Cn(A) \div \beta$.
Case 2) $\alpha, \beta \in Cn(A)$ and $\mathit{cut}_{\preceq}(\beta) \subseteq \mathit{cut}_{\preceq}(\alpha)$.
Case 2.1) $\vdash \beta$. Trivial.
Case 2.2) $\nvdash \beta$.
$\{\gamma \in A : \{\delta \in A : \gamma \prec \delta\} \nvdash \beta\} \subseteq \{\gamma \in A : \{\delta \in A : \gamma \prec \delta\} \nvdash \alpha\}$.
Hence,
$\{\gamma \in A : \{\delta \in A : (\gamma \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \gamma$ and $\nvdash \gamma)$ or $(\vdash \delta$ and $\delta \in Cn(A) \div \gamma$ and $\nvdash \gamma)\} \nvdash \beta\} \subseteq \{\gamma \in A : \{\delta \in A : (\gamma \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \gamma$ and $\nvdash \gamma)$ or $(\vdash \delta$ and $\delta \in Cn(A) \div \gamma$ and $\nvdash \gamma)\} \nvdash \alpha\}$.
Therefore according to $(\div 1)$ and $(\div 4)$,
$X = \{\gamma \in A : \{\delta \in A : (\gamma \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \gamma)$ or $(\vdash \delta$ and $\nvdash \gamma)\} \nvdash \beta\} \subseteq Y = \{\gamma \in A : \{\delta \in A : (\gamma \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \gamma)$ or $(\vdash \delta$ and $\nvdash \gamma)\} \nvdash \alpha\}$. Assume by *reductio* that $\alpha \in Cn(A) \div \beta$. From $\alpha \in Cn(A) \div \beta$ it follows, by *base-reduction*, that $Cn(A) \div \beta \cap A \vdash \alpha$. By compactness, there exists a finite subset of $Cn(A) \div \beta \cap A$, $H = \{\alpha_1, ..., \alpha_n\}$, such that $H \vdash \alpha$. Let us assume that $H \cap Cn(\emptyset) = \emptyset$. For all $\alpha_i \in H$, $\alpha_i \in Cn(A) \div \beta = Cn(A) \div \beta'$, for some $\beta' \in A$ (by Observation 17). Hence, by *expulsiveness*, $\beta' \notin Cn(A) \div \alpha_i$. Therefore $\beta' \notin Y$, since $H \subseteq Z = \{\delta \in A : (\beta' \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \beta')$ or $(\vdash \delta$ and $\nvdash \beta')\}$. On the other hand $\beta' \in X$, since $Z \subseteq Cn(A) \div \beta'$, and by $(\div 4)$ $Cn(A) \div \beta' \nvdash \beta$. Hence $X \nsubseteq Y$. Contradiction.
(⇐) Let $\alpha \notin Cn(A) \div \beta$ or $\vdash \beta$. We will prove by cases:
Case 1) $\alpha \notin Cn(A)$. Trivial.
Case 2) $\alpha \in Cn(A)$.
Case 2.1) $\vdash \beta$. Then $\alpha, \beta \in Cn(A)$ and $\mathit{cut}_{\preceq}(\beta) \subseteq \mathit{cut}_{\preceq}(\alpha)$.
Case 2.2) $\alpha \notin Cn(A) \div \beta$ and $\nvdash \beta$. Hence, it follows that $\beta \in Cn(A)$, $\nvdash \alpha$ and $Cn(A) \div \beta \subseteq Cn(A) \div \alpha$, by $(\div 3)$, $(\div 1)$ and $(\div 9)$, respectively. Let us assume by *reductio* that $\mathit{cut}_{\preceq}(\beta) \nsubseteq \mathit{cut}_{\preceq}(\alpha)$. Hence there exists $\psi \in A$ such that $\psi \in \mathit{cut}_{\preceq}(\beta)$ and $\psi \notin \mathit{cut}_{\preceq}(\alpha)$. From which follows that $\nvdash \psi$, $C = \{\delta \in A : (\psi \notin Cn(A) \div \delta$ and $\delta \in Cn(A) \div \psi)$ or $(\vdash \delta$ and $\nvdash \psi)\} \nvdash \beta$ and $C \vdash \alpha$. $C \subseteq Cn(A) \div \psi$. Then $Cn(A) \div \psi \vdash \alpha$. Hence, by $(\div 4)$ and *linearity*, it follows that $Cn(A) \div \alpha \subset Cn(A) \div \psi$. From $Cn(A) \div \beta \subseteq Cn(A) \div \alpha$ it follows that $Cn(A) \div \beta \subset Cn(A) \div \psi$. By $(\div 9)$, $\beta \in Cn(A) \div \psi$. Therefore, by *base-reduction*, $Cn(A) \div \psi \cap A \vdash \beta$. On the other hand $Cn(A) \div \psi \cap A \subseteq C$. Hence $C \vdash \beta$. Contradiction. ■

References

Alchourrón, C., and Makinson, D. 1985. On the logic of theory change: Safe contraction. *Studia Logica* 44:405–422.

Alchourrón, C.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50:510–530.

Arló-Costa, H., and Levi, I. 2006. Contraction: On the decision-theoretical origins of minimal change and entrenchment. *Synthese* 152:1:129–154.

Fermé, E., and Hansson, S. O. 2011. AGM 25 years: Twenty-five years of research in belief change. *Journal of Philosophical Logic* 40:295–331.

Fermé, E., and Rodriguez, R. 1998. A brief note about the Rott contraction. *Logic Journal of the IGPL* 6(6):835–842.

Fermé, E.; Krevneris, M.; and Reis, M. 2008. An axiomatic characterization of ensconcement-based contraction. *Journal of Logic and Computation* 18(5):739–753.

Fermé, E. 1998. On the logic of theory change: Contraction without recovery. *Journal of Logic, Language and Information* 7:127–137.

Fuhrmann, A. 1991. Theory contraction through base contraction. *Journal of Philosophical Logic* 20:175–203.

Garapa, M.; Fermé, E.; and Reis, M. D. L. 2016. Ensconcement and contraction. (unpublished manuscript).

Gärdenfors, P., and Makinson, D. 1988. Revisions of knowledge systems using epistemic entrenchment. In Vardi, M. Y., ed., *Proceedings of the Second Conference on Theoretical Aspects of Reasoning About Knowledge*, 83–95. Los Altos: Morgan Kaufmann.

Gärdenfors, P. 1982. Rules for rational changes of belief. In Pauli, T., ed., *Philosophical Essays dedicated to Lennart Åqvist on his fiftieth birthday*, number 34 in Philosophical Studies, 88–101.

Gärdenfors, P. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Cambridge: The MIT Press.

Grove, A. 1988. Two modellings for theory change. *Journal of Philosophical Logic* 17:157–170.

Hansson, S. O. 1991. Belief contraction without recovery. *Studia Logica* 50:251–260.

Hansson, S. O. 1994. Kernel contraction. *Journal of Symbolic Logic* 59:845–859.

Hansson, S. O. 1999a. Revision of belief sets and belief bases. In Dubois, D., and Prade, H., eds., *Belief Change*, Handbook of Defeasible Reasoning and Uncertainty Management Systems. Dordrecht: Springer Netherlands. 17–75.

Hansson, S. O. 1999b. *A Textbook of Belief Dynamics. Theory Change and Database Updating.* Applied Logic Series. Dordrecht: Kluwer Academic Publishers.

Levi, I. 1991. *The fixation of belief and its undoing: changing beliefs through inquiry.* Cambridge: Cambridge University Press.

Lindström, S., and Rabinowicz, W. 1991. Epistemic entrenchment with incomparabilities and relational belief revision. In Fuhrmann, and Morreau., eds., *The Logic of Theory Change*, 93–126. Berlin: Springer-Verlag.

Niederée, R. 1991. Multiple contraction: A further case against Gärdenfors' principle of recovery. In Fuhrmann, and Morreau., eds., *The Logic of Theory Change*, 322–334. Berlin: Springer-Verlag.

Rott, H., and Pagnucco, M. 1999. Severe withdrawal (and recovery). *Journal of Philosophical Logic* 28:501–547.

Rott, H. 1991. Two methods of constructing contractions and revisions of knowledge systems. *Journal of Philosophical Logic* 20:149–173.

Rott, H. 1995. “Just because”. Taking belief bases very seriously. In Hansson, S. O., and Rabinowicz, W., eds., *Logic for a change*, number 9 in Uppsala Prints and Preprints in Philosophy. Dep. of Philosophy, Uppsala University. 106–124.

Williams, M.-A. 1992. Two operators for theory bases. In *Proc. Australian Joint Artificial Intelligence Conference*, 259–265. World Scientific.

Williams, M.-A. 1994a. On the logic of theory base change. In MacNish., ed., *Logics in Artificial Intelligence*, number 835 in Lecture Notes Series in Computer Science. Springer Verlag.

Williams, M.-A. 1994b. Transmutations of knowledge systems. In Doyle, J.; Sandewall, E.; and Torasso, P., eds., *Proceedings of the fourth International Conference on Principles of Knowledge Representation and Reasoning*. Bonn, Germany: Morgan Kaufmann. 619–629.

Williams, M.-A. 1995. Iterated theory base change: A computational model. In *Proc. of the 14th IJCAI*, 1541–1547.

A strengthening of rational closure in DLs: reasoning about multiple aspects

Valentina Gliozzi
Center for Logic, Language and Cognition
Dipartimento di Informatica - Università di Torino - Italy
valentina.gliozzi@unito.it

Abstract

We propose a logical analysis of the concept of typicality, central in human cognition (Rosch,1978). We start from a previously proposed extension of the basic Description Logic $\mathcal{ALC}$ with a typicality operator $\mathbf{T}$ that allows to consistently represent the attribution to classes of individuals of properties with exceptions (as in the classic example (i)typical birds fly, (ii) penguins are birds but (iii)typical penguins don't fly). We then strengthen this extension in order to separately reason about the typicality with respect to different aspects (e.g., flying, having nice feather: in the previous example, penguins may not inherit the property of flying, for which they are exceptional, but can nonetheless inherit other properties, such as having nice feather).

Introduction

In (Giordano et al. 2015) it is proposed a rational closure strengthening of $\mathcal{ALC}$. This strengthening allows to perform non monotonic reasoning in $\mathcal{ALC}$ in a computationally efficient way. The extension, as already the related logic $\mathcal{ALC} + \mathbf{T}_{min}$ proposed in (Giordano et al. 2013a) and the weaker (monotonic) logic $\mathcal{ALC} + \mathbf{T}$ presented in (Giordano et al. 2009), allows to consistently represent typical properties with exceptions that could not be represented in standard $\mathcal{ALC}$.

For instance, in all the above logics one can say that:

SET 1:
Typical students don't earn money
Typical working students do earn money
Typical apprentice working students don't earn money

without having to conclude that there cannot exist working students nor apprentice working students. On the contrary, in standard $\mathcal{ALC}$ typicality cannot be represented, and these three propositions can only be expressed by the stronger ones:

SET 2:
Students don't earn money (Student $\sqsubseteq$ $\neg$ EarnMoney)
Working students do earn money (Worker $\sqcap$ Student $\sqsubseteq$ EarnMoney)
Apprentice working students don't earn money (Worker $\sqcap$ Apprentice $\sqcap$ Student $\sqsubseteq$ $\neg$ EarnMoney)

These propositions are consistent in $\mathcal{ALC}$ only if there are no working students nor apprentice working students.

In all the extensions of $\mathcal{ALC}$ mentioned above one can represent the set of propositions in $SET1$ by means of a typicality operator $\mathbf{T}$ that, given a concept C (e.g. Student) singles out the most typical instances of C: so, for instance, $\mathbf{T}(Student)$ refers to the typical instances of the concept Student. The semantics of $\mathbf{T}$ is given by means of a preference relation $<$ that compares the typicality of two individuals: for any two x and y, $x < y$ means that x is more typical than y. Typical instances of a concept C are those minimal with respect to $<$ (formally, as we will see later, $(\mathbf{T}(C))^I = min_<(C)^I$, where $min_<(C)^I = \{x \in C^I : \nexists y \in C^I$ s.t. $y < x\}$).

The operator $\mathbf{T}$ has all the properties that, in the analysis of Kraus Lehmann and Magidor (Kraus, Lehmann, and Magidor 1990) any non monotonic entailment should have. For instance, $\mathbf{T}$ satisfies the principle of cautious monotonicity, according to which if $\mathbf{T}(Student) \sqsubseteq Young$, then $\mathbf{T}(Student) = \mathbf{T}(Student \sqcap Young)$). The precise relations between the properties of $\mathbf{T}$ and preferential entailment are established in (Giordano et al. 2009).

Although the extensions of $\mathcal{ALC}$ with the typicality operator $\mathbf{T}$ allow to express $SET1$ of propositions, the resulting logic is monotonic, and it does not allow to perform some wanted, non monotonic inferences. For instance, it does not allow to deal with irrelevance which is the principle that from the fact that typical students are young, one would want to derive that typical blond students also are young, since being blond is irrelevant with respect to youth. As another example, when knowing that an individual, say John, is a student, and given $SET1$ of propositions, one would want to conclude that John is a typical student and therefore does not earn money. On the other hand, when knowing that John is a working student, one would want to conclude that he is a typical working student and therefore does earn money. In other words one would want to assume that an individual is a typical instance of the most specific class it belongs to, in the absence of information to the contrary.

These stronger inferences all hold in the strengthening of $\mathcal{ALC} + \mathbf{T}$ presented in (Giordano et al. 2013a; 2015). In particular, (Giordano et al. 2015) proposes an adaptation to $\mathcal{ALC}$ of the well known mechanism of *rational closure*, first proposed by Lehman and Magidor in (Lehmann and Magidor 1992). From a semantic point of view, this strengthening of $\mathcal{ALC} + \mathbf{T}$ corresponds to restricting one's attention to minimal models, that minimize the height (rank) of all domain elements with respect to $<$ (i.e. that minimize the length of the $<$-chains starting from all individuals). Under the condition that the models considered are canonical, the semantic characterization corresponds to the syntactical rational closure. This semantics supports all the above wanted inferences, and the nice computational properties of rational closure guarantee that whether the above inferences are valid or not can be computed in reasonable time.

The main drawback of rational closure is that it is an **all-or-nothing** mechanism: for any subclass C' of C it holds that either the typical members of C' inherit all the properties of C or they don't inherit any property. Once the typical members of C' are recognized as exceptional with respect to C for a given aspect, they become exceptional for all aspects. Consider the classic birds/penguins example, expressed by propositions:

SET 3:
Typical birds have nice feather
Typical birds fly
Penguins are birds
Typical penguins do not fly

In this case, since penguins are exceptional with respect to the aspect of flying, they are *non-typical* birds, and for this reason they do not inherit any of the typical properties of birds.

On the contrary, given $SET3$ of propositions, one wants to conclude that:

- **(**) Typical penguins have nice feather**

This is to say that one wants to separately reason about the different aspects: the property of flying is not related to the property of having nice feather, hence we want to separately reason on the two aspects.

Here we propose a strengthening of the semantics used for rational closure in $\mathcal{ALC}$ (Giordano et al. 2015) that only used a single preference relation $<$ by allowing, beside $<$, several preference relations that compare the typicality of individuals with respect to a given aspect. **Obtaining a strengthening of rational closure is the purpose of this work. This puts strong constraints on the resulting semantics, and defines the horizon of this work**. In this new semantics we can express the fact that, for instance, x is more typical than y with respect to the property of flying but y is more typical that x with respect to some other property, as the property of having nice feather. To this purpose we consider preference relations indexed by concepts that stand for the above mentioned aspects under which we compare individuals. So we will write that $x <_A y$ to mean that x is preferred to y for what concerns aspect A: for instance $x <_{Fly} y$ means that x is more typical than y with respect to the property of flying.

We therefore proceed as follows: we first recall the semantics of the extension of $\mathcal{ALC}$ with a typicality operator which was at the basis of the definition of rational closure and semantics in (Giordano et al. 2013b; 2015). We then expand this semantics by introducing several preference relations, that we then minimize obtaining our new minimal models' mechanism. As we will see this new semantics leads to a strengthening of rational closure, allowing to separately reason about the inheritance of different properties.

The operator T and the General Semantics

Let us briefly recall the logic $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ which is at the basis of a rational closure construction proposed in (Giordano et al. 2015) for $\mathcal{ALC}$. The intuitive idea of $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ is to extend the standard $\mathcal{ALC}$ with concepts of the form $\mathbf{T}(C)$, whose intuitive meaning is that $\mathbf{T}(C)$ selects the *typical* instances of a concept C, to distinguish between the properties that hold for all instances of concept C ($C \sqsubseteq D$), and those that only hold for the typical such instances ($\mathbf{T}(C) \sqsubseteq D$). The $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ language is defined as follows: $C_R := A \mid \top \mid \bot \mid \neg C_R \mid C_R \sqcap C_R \mid C_R \sqcup C_R \mid \forall R.C_R \mid \exists R.C_R$, and $C_L := C_R \mid \mathbf{T}(C_R)$, where A is a concept name and R a role name. A KB is a pair (TBox, ABox). TBox contains a finite set of concept inclusions $C_L \sqsubseteq C_R$. ABox contains a finite set of assertions of the form $C_L(a)$ and $R(a, b)$, where a, b are individual constants.

The semantics of $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ is defined in terms of rational models: ordinary models of $\mathcal{ALC}$ are equipped with a *preference relation* $<$ on the domain, whose intuitive meaning is to compare the "typicality" of domain elements: $x < y$ means that x is more typical than y. Typical members of a concept C, instances of $\mathbf{T}(C)$, are the members x of C that are minimal with respect to $<$ (such that there is no other member of C more typical than x). In rational models $<$ is further assumed to be modular: for all $x, y, z \in \Delta$, if $x < y$ then either $x < z$ or $z < y$. These rational models characterize $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$.

Definition 1 (Semantics of $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ (Giordano et al. 2015)) *A model $\mathcal{M}$ of $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ is any structure $\langle \Delta, <, I \rangle$ where: Δ is the domain; $<$ is an irreflexive, transitive, and modular relation over Δ that satisfies the* finite chain condition*(there is no infinite $<$-descending chain, hence if $S \neq \emptyset$, also $min_<(S) \neq \emptyset$); I is the extension function that maps each concept name C to $C^I \subseteq \Delta$, each role name R to $R^I \subseteq \Delta^I \times \Delta^I$ and each individual constant $a \in \mathcal{O}$ to $a^I \in \Delta$. For concepts of $\mathcal{ALC}$, C^I is defined in the usual way. For the* $\mathbf{T}$ *operator, we have $(\mathbf{T}(C))^I = min_<(C^I)$.*

As shown in (Giordano et al. 2015), the logic $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ enjoys the finite model property and finite $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ models can be equivalently defined by postulating the existence of a function $k_\mathcal{M} : \Delta \longmapsto \mathbb{N}$, where $k_\mathcal{M}$ assigns a finite rank to each world: the rank $k_\mathcal{M}$ of a domain element $x \in \Delta$ is the length of the longest chain $x_0 < \cdots < x$ from x to a minimal x_0 (s. t. there is no x' with $x' < x_0$). The rank $k_\mathcal{M}(C_R)$ of a concept C_R in $\mathcal{M}$ is $i = min\{k_\mathcal{M}(x) : x \in C_R^I\}$.

A model $\mathcal{M}$ satisfies a knowledge base K=(TBox,ABox) if it satisfies its TBox (and for all inclusions $C \sqsubseteq D$ in TBox, it holds $C^I \subseteq D^I$), and its ABox (for all $C(a)$ in ABox, $a^I \in C^I$, and for all aRb in ABox, $(a^I, b^I) \in R^I$). A query F (either an assertion $C_L(a)$ or an inclusion relation $C_L \sqsubseteq C_R$) is logically (rationally) entailed by a knowledge base K ($K \models_{\mathcal{ALC}+\mathbf{T}_\mathsf{R}} F$) if F holds in all models satisfying K.

Although the typicality operator $\mathbf{T}$ itself is nonmonotonic (i.e. $\mathbf{T}(C) \sqsubseteq D$ does not imply $\mathbf{T}(C \sqcap E) \sqsubseteq D$), the logic $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$ is monotonic: what is logically entailed by K is still entailed by any K' with $K \subseteq K'$.

In (Giordano et al. 2013b; 2015) the non monotonic mechanism of rational closure has been defined over $\mathcal{ALC} + \mathbf{T}_\mathsf{R}$, which extends to DLs the notion of rational closure proposed in the propositional context by Lehmann and Magidor (Lehmann and Magidor 1992). The definition is based on the notion of exceptionality. Roughly speaking $\mathbf{T}(C) \sqsubseteq D$ holds (is included in the rational closure) of K if C (indeed, $C \sqcap D$) is less exceptional than $C \sqcap \neg D$. We briefly recall this construction and we refer to (Giordano et al. 2013b; 2015) for full details. Here we only consider rational closure of TBox, defined as follows.

Definition 2 (Exceptionality of concepts and inclusions) *Let T_B be a TBox and C a concept. C is said to be* exceptional *for T_B if and only if $T_B \models_{\mathcal{ALC}+\mathbf{T}_R} \mathbf{T}(\top) \sqsubseteq \neg C$. A $\mathbf{T}$-inclusion $\mathbf{T}(C) \sqsubseteq D$ is exceptional for T_B if C is exceptional for T_B. The set of $\mathbf{T}$-inclusions of T_B which are exceptional in T_B will be denoted as $\mathcal{E}(T_B)$.*

Given a DL TBox, it is possible to define a sequence of non increasing subsets of TBox ordered according to the exceptionality of the elements $E_0 \supseteq E_1, E_1 \supseteq E_2, \dots$ by letting $E_0 =$ TBox and, for $i > 0$, $E_i = \mathcal{E}(E_{i-1}) \cup \{C \sqsubseteq D \in$ TBox s.t. $\mathbf{T}$ does not occurr in $C\}$. Observe that, being KB finite, there is an $n \geq 0$ such that, for all $m > n, E_m = E_n$ or $E_m = \emptyset$. A concept C has *rank* i (denoted $rank(C) = i$) for TBox, iff i is the least natural number for which C is not exceptional for E_i. If C is exceptional for all E_i then $rank(C) = \infty$ (C has no rank).

Rational closure builds on this notion of exceptionality:

Definition 3 (Rational closure of TBox) *Let KB = (TBox, ABox) be a DL knowledge base. The rational closure of TBox $\overline{TBox} = \{\mathbf{T}(C) \sqsubseteq D \mid$ either $rank(C) < rank(C \sqcap \neg D)$ or $rank(C) = \infty\} \cup \{C \sqsubseteq D \mid KB \models_{\mathcal{ALC}+\mathbf{T}_R} C \sqsubseteq D\}$, where C and D are $\mathcal{ALC}$ concepts.*

As a very interesting property, in the context of DLs, the rational closure has a very interesting complexity: deciding if an inclusion $\mathbf{T}(C) \sqsubseteq D$ belongs to the rational closure of TBox is a problem in ExpTime (Giordano et al. 2015).

In (Giordano et al. 2015) it is shown that the semantics corresponding to rational closure can be given in terms of minimal canonical $\mathcal{ALC} + \mathbf{T}_R$ models. With respect to standard $\mathcal{ALC} + \mathbf{T}_R$ models, in these models the rank of each domain element is as low as possible (each domain element is assumed to be as typical as possible). This is expressed by the following definition.

Definition 4 (Minimal models of K (with respect to $TBox$)) *Given $\mathcal{M}$ $=\langle\Delta, <, I\rangle$ and $\mathcal{M}' = \langle\Delta', <', I'\rangle$, we say that $\mathcal{M}$ is preferred to $\mathcal{M}'$ ($\mathcal{M} < \mathcal{M}'$) if: $\Delta = \Delta'$, $C^I = C^{I'}$ for all concepts C, for all $x \in \Delta$, it holds that $k_{\mathcal{M}}(x) \leq k_{\mathcal{M}'}(x)$ whereas there exists $y \in \Delta$ such that $k_{\mathcal{M}}(y) < k_{\mathcal{M}'}(y)$.*

Given a knowledge base $K = \langle TBox, ABox\rangle$, we say that $\mathcal{M}$ is a minimal model of K (with respect to TBox) if it is a model satisfying K and there is no $\mathcal{M}'$ model satisfying K such that $\mathcal{M}' < \mathcal{M}$.

Furthermore, the models corresponding to rational closure are canonical. This property, expressed by the following definition, is needed when reasoning about the (relative) rank of the concepts: it is important to have them all represented.

Definition 5 (Canonical model) *Given K=(TBox,ABox),a model $\mathcal{M}$ $=\langle\Delta, <, I\rangle$ satisfying K is* canonical *if for each set of concepts $\{C_1, C_2, \dots, C_n\}$consistent with K, there exists (at least) a domain element $x \in \Delta$ such that $x \in (C_1 \sqcap C_2 \sqcap \dots \sqcap C_n)^I$.*

Definition 6 (Minimal canonical models (with respect to TBox)) *$\mathcal{M}$ is a canonical model of K minimal with respect to TBox if it satisfies K, it is minimal with respect to $TBox$ (Definition 4) and it is canonical (Definition 5).*

The correspondence between minimal canonical models and rational closure is established by the following key theorem.

Theorem 1 ((Giordano et al. 2015)) *Let K=(TBox,ABox) be a knowledge base and $C \sqsubseteq D$ a query. We have that $C \sqsubseteq D \in \overline{TBox}$ if and only if $C \sqsubseteq D$ holds in all minimal canonical models of K with respect to TBox (Definition 6).*

Semantics with several preference relations

The main weakness of rational closure, despite its power and its nice computational properties, is that it is an all-or-nothing mechanism that does not allow to separately reason on single aspects. To overcome this difficulty, we here consider models with several preference relations, one for each aspect we want to reason about. We assume this is any concept occurring in K: we call $\mathcal{L}_{\mathcal{A}}$ the set of these aspects (observe that A may be non-atomic). For each aspect A, $<_A$ expresses the preference for aspect A : $<_{Fly}$ expresses the preference for flying, so if we know that $\mathbf{T}(Bird) \sqsubseteq Fly$, birds that do fly will be preferred to birds that do not fly, with respect to aspect fly, i.e. with respect to $<_{Fly}$. All these preferences, as well as the global preference relation $<$, satisfy the properties in Definition 7 below. We now enrich the definition of an $\mathcal{ALC} + \mathbf{T}_R$ model given above (Definition 1) by taking into account preferences with respect to all of the aspects. In the semantics we can express that for instance $x <_{A_i} y$, whereas $y <_{A_j} x$ (x is preferred to y for aspect A_i but y is preferred to x for aspect A_j).

This semantic richness allows to obtain a strengthening of rational closure in which typicality with respect to every aspect is maximized. Since we want to compare our approach to rational closure, we keep the language the same than in $\mathcal{ALC} + \mathbf{T}_R$. In particular, we only have one single typicality operator $\mathbf{T}$. However, the semantic richness could motivate the introduction of several typicality operators $\mathbf{T}_{A_1} \dots \mathbf{T}_{A_n}$ by which one might want to explicitly talk in the language about the typicality w.r.t. aspect A_1, or A_2, and so on. We leave this extension for future work.

Definition 7 (Enriched rational models) *Given a knowledge base K, we call an enriched rational model a structure $\mathcal{M} = \langle\Delta, <, <_{A_1}, \dots, <_{A_n}, I\rangle$, where Δ, I are defined as in Definition 1, and $<, <_{A_1}, \dots, <_{A_n}$ are preference relations over Δ, with the properties of being irreflexive, transitive, satisfying the finite chain condition, modular (for all $x, y, z \in \Delta$, if $x <_{A_i} y$ then either $x <_{A_i} z$ or $z <_{A_i} y$).*

For all $<_{A_i}$ and for $<$ it holds that $min_{<_{A_i}}(S) = \{x \in S$ s.t. there is no $x_1 \in S$ s.t. $x_1 <_{A_i} x\}$ and $min_{<}(S) =$

$\{x \in S$ *s.t. there is no* $x_1 \in S$ *s.t.* $x_1 < x\}$ *and* $(\mathbf{T}(C))^I = min_<(C^I)$.

$<$ *satisfies the further conditions that* $x < y$ *if:*
(a) there is A_i *such that* $x <_{A_i} y$*, and there is no* A_j *such that* $y <_{A_j} x$ *or;*
(b) there is $\mathbf{T}(C_i) \sqsubseteq A_i \in K$ *s.t.* $y \in (C_i \sqcap \neg A_i)^I$*, and for all* $\mathbf{T}(C_j) \sqsubseteq A_j \in K$ *s.t.* $x \in (C_j \sqcap \neg A_j)^I$ *, there is* $\mathbf{T}(C_k) \sqsubseteq A_k \in K$ *s.t.* $y \in (C_k \sqcap \neg A_k)^I$ *and* $k_{\mathcal{M}}(C_j) < k_{\mathcal{M}}(C_k)$.

In this semantics the global preference relation $<$ is related to the various preference relations $<_{A_i}$ relative to single aspects A_i. Given (a) $x < y$ when x is preferred to y for a single aspect A_i, and there is no aspect A_j for which y is preferred to x. (b) captures the idea that in case two individuals are preferred with respect to different aspects, preference (for the global preference relation) is given to the individual that satisfies all typical properties of the **most specific** concept (if C_k is more specific than C_j, then $k_{\mathcal{M}}(C_j) < k_{\mathcal{M}}(C_k)$), as illustrated by Example 1 below.

We insist in highlighting that this semantics somewhat complicated is needed since we want to provide a strengthening of rational closure. For this, we have to respect the constraints imposed by rational closure. One might think in the future to study a semantics in which only (a) holds.We have not considered such a simpler semantics since it would no longer be a strengthening of the semantics corresponding to rational closure, and is therefore out of the focus of this work.

In order to be a model of K an $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ model must satisfy the following constraints.

Definition 8 (Enriched rational models of K) *Given a knowledge base K, and an enriched rational model for K* $\mathcal{M} = \langle \Delta, <, <_{A_1}, \ldots, <_{A_n}, I \rangle$*,* $\mathcal{M}$ *is a model of K if it satisfies both its TBox and its ABox, where* $\mathcal{M}$ *satisfies TBox if for all inclusions* $C \sqsubseteq A_i \in TBox$*: if* $\mathbf{T}$ *does not occur in* C*, then* $C^I \subseteq {A_i}^I$ *if* $\mathbf{T}$ *occurs in* C*, and* C *is* $\mathbf{T}(C')$*, then both (i)* $min_<({C'}^I) \subseteq {A_i}^I$ *and (ii)* $min_{<_{A_i}}({C'}^I) \subseteq {A_i}^I$*.* $\mathcal{M}$ *satisfies ABox if (i) for all* $C(a)$ *in ABox,* $a^I \in C^I$*, (ii) for all* aRb *in ABox,* $(a^I, b^I) \in R^I$

Example 1 *Let* $K = \{Penguin \sqsubseteq Bird, \mathbf{T}(Bird) \sqsubseteq HasNiceFeather,\ \mathbf{T}(Bird) \sqsubseteq Fly,\ \mathbf{T}(Penguin) \sqsubseteq \neg Fly\}$. $\mathcal{L}_A = \{HasNiceFeather, Fly, \neg Fly, Bird, Penguin\}$*. We consider an* $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ *model* $\mathcal{M}$ *of K, that we don't fully describe but which we only use to observe the behavior of two Penguins x, y with respect to the properties of (not) flying and having nice feather. In particular, let us consider the three preference relations:* $<, <_{\neg Fly}, <_{HasNiceFeather}$.

Suppose $x <_{\neg Fly} y$ *(because* x*, as all typical penguins, does not fly whereas* y *exceptionally does) and there is no other aspect* A_i *such that* $y <_{A_i} x$*, and in particular it does not hold that* $y <_{HasNiceFeather} x$ *(because for instance both have a nice feather). In this case, obviously it holds that* $x < y$ *(since (a) is satisfied).*

Consider now a more tricky situation in which again $x <_{\neg Fly} y$ *holds (because for instance* x *does not fly whereas* y *flies), (*x *is a typical penguin for what concerns Flying) but this time* $y <_{HasNiceFeather} x$ *holds (because for instance* y *has a nice feather, whereas* x *has not). So* x *is preferred to* y *for a given aspect whereas* y *is preferred to* x *for another aspect. However,* x *enjoys the typical properties of penguins, and violates the typical properties of birds, whereas* y *enjoys the typical properties of birds and violates those of penguins. Being concept Penguin more specific than concept Bird, we prefer* x *to* y*, since we prefer the individuals that inherit the properties of the most specific concepts of which they are instances. This is exactly what we get: by (b)* $x < y$ *holds.*

Logical entailment for $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ is defined as usual: a query (with form $C_L(a)$ or $C_L \sqsubseteq C_R$) is logically entailed by K if it holds in all models of K, as stated by the following definition. The following theorem shows the relations between $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ and $\mathcal{ALC} + \mathbf{T}_{\mathsf{R}}$. Proofs are omitted due to space limitations.

Theorem 2 *If* $K \models_{\mathcal{ALC}+\mathbf{T}_R} F$ *then also* $K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E} F$*. If* $\mathbf{T}$ *does not occur in* F *the other direction also holds: If* $K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E} F$ *then also* $K \models_{\mathcal{ALC}+\mathbf{T}_R} F$*.*

The following example shows that $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ alone is not strong enough, and this motivates the minimal models' mechanism that we introduce in the next section. In the example we show that $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ alone does not allow us to perform the stronger inferences with respect to rational closure mentioned in the Introduction (and in particular, it does not allow to infer (**), that typical penguins have a nice feather).

Example 2 *Consider the above Example 1. As said in the Introduction, in rational closure we are not able to reason separately about the property of flying or not flying, and the property of having or not having a nice feather. Since penguins are exceptional birds with respect to the property of flying, in rational closure which is an all-or-nothing mechanism, they do not inherit any of the properties of typical birds. In particular, they do not inherit the property of having a nice feather, even if this property and the fact of flying are independent from each other and there is no reason why being exceptional with respect to one property should block the inheritance of the other one. Does our enriched semantics enforce the separate inheritance of independent properties?*

Consider a model $\mathcal{M}$ *in which we have* $\Delta = \{x, y, z\}$*, where* x *is a bird (not a penguin) that flies and has a nice feather (*$x \in Bird^I, x \in Fly^I, x \in HasNiceFeather^I, x \notin Penguin^I$*),* y *is a penguin that does not fly and has a nice feather (*$y \in Penguin^I, y \in Bird^I, y \notin Fly^I, y \in HasNiceFeather^I$*),* z *is a penguin that does not fly and has no nice feather (*$z \in Penguin^I, z \in Bird^I, z \notin Fly^I, z \notin HasNiceFeather^I$*). Suppose it holds that* $x <_{Fly} y$*,* $x <_{Fly} z$*,* $x <_{HasNiceFeather} z$*,* $y <_{HasNiceFeather} z$*, and* $x < y$*,* $x < z$*,* $y < z$*. It can be verified that this is an* $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ *model, satisfying* $\mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$ *(since the only typical Penguin is* y*, instance of HasNiceFeather).*

Unfortunately, this is not the only $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ *model of* K*. For instance there can be* $\mathcal{M}'$ *equal to* $\mathcal{M}$ *except from the fact that* $y <_{HasNiceFeather} z$ *does not hold, nor* $y < z$ *holds. It can be easily verified that this is also an* $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ *model of* K *in which* $\mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$

does not hold (since now also z is a typical Penguin, and z is not an instance of HasNiceFeather).

This example shows that although there are $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ models satisfying well suited inclusions, the logic is not strong enough to limit our attention to these models. We would like to constrain our logic in order to exclude models like $\mathcal{M}'$. Roughly speaking, we want to eliminate $\mathcal{M}'$ because it is not minimal: although the model as it is satisfies K, so y does not *need* to be preferred to z to satisfy K (neither with respect to $<$ nor with respect to $<_{HasNiceFeather}$), intuitively we would like to prefer y to z (with respect to the property HasNiceFeather, whence in this case with respect to the global $<$), since y does not falsify any of the inclusions with HasNiceFeather, whereas z does. This is obtained by imposing the constraint of considering only models minimal with respect to all relations $<_A$, defined as in Definition 10 below. Notice that the wanted inference does not hold in $\mathcal{ALC} + \mathbf{T}_{\mathsf{R}}$ minimal canonical models corresponding to rational closure: in these models $y < z$ does never hold (the two elements have the same rank) and this semantics does not allow us to prefer y to z. By adopting the restriction to minimal canonical models, we obtain a semantics which is stronger than rational closure (and therefore enforces all conclusions enforced by rational closure) and, furthermore, separately allows to reason on different aspects.

Before we end the section, similarly to what done above, let us introduce a rank of a domain element with respect to an aspect. We will use this notion in the following section.

Definition 9 *The rank $k_{A_{i\mathcal{M}}}(x)$ of a domain element x with respect to $<_{A_i}$ in $\mathcal{M}$ is the length of the longest chain $x_0 <_{A_i} \cdots <_{A_i} x$ from x to a minimal x_0 (s.t. for no x' $x' <_{A_i} x_0$). To refer to the rank of an element x with respect to the preference relation $<$ we will simply write $k_{\mathcal{M}}(x)$.*

The notion just introduced will be useful in the following. Since $k_{A_{i\mathcal{M}}}$ and $<_{A_i}$ are clearly interdefinable (by the previous definition and by the properties of $<_{A_i}$ it easily follows that in all enriched models $\mathcal{M}$, $x <_{A_i} y$ iff $k_{A_{i\mathcal{M}}}(x) < k_{A_{i\mathcal{M}}}(y)$, and $x < y$ iff $k_{\mathcal{M}}(x) < k_{\mathcal{M}}(y)$), we will shift from one to other whenever this simplifies the exposition.

Nonmonotonicity and relation with rational closure

We here define a minimal models mechanism starting from the enriched models of the previous section. With respect to the minimal canonical models used in (Giordano et al. 2015) we define minimal models by separately minimizing all the preference relations with respect to all aspects (steps (i) and (ii) in the definition below), before minimizing $<$ (steps (iii) and (iv) in the definition below). By the constraints linking $<$ to the preference relations $<_{A_1} \cdots <_{A_n}$, this leads to preferring (with respect to the global $<$) the individuals that are minimal with respect to all $<_{A_i}$ for all aspects A_i, or to aspects of most specific categories than of more general ones. It turns out that this leads to a stronger semantics than what is obtained by directly minimizing $<$.

Definition 10 (Minimal Enriched Models) *Given two $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ enriched models $\mathcal{M} = \langle \Delta, <_{A_1}, \ldots, <_{A_n}, <, I \rangle$ and $\mathcal{M}' = \langle \Delta', <'_{A_1}, \ldots, <'_{A_n}, <', I' \rangle$ we say that $\mathcal{M}'$ is preferred to $\mathcal{M}$ with respect to the single aspects (and write $\mathcal{M}' <_{Enriched_Aspects} \mathcal{M}$) if $\Delta = \Delta'$, $I = I'$, and:*

- *(i) for all $x \in \Delta$, for all A_i: $k_{A_{i\mathcal{M}'}}(x) \leq k_{A_{i\mathcal{M}}}(x)$;*
- *(ii) for some $y \in \Delta$, for some A_j, $k_{A_{j\mathcal{M}'}}(y) < k_{A_{j\mathcal{M}}}(y)$*

We let the set $Min_{Aspects} = \{\mathcal{M}$: there is no $\mathcal{M}'$ such that $\mathcal{M}' <_{Enriched_Aspects} \mathcal{M}\}$.
Given $\mathcal{M}$ and $\mathcal{M}' \in Min_{Aspects}$, we say that $\mathcal{M}'$ is overall preferred to $\mathcal{M}$ (and write $\mathcal{M}' <_{Enriched} \mathcal{M}$) if$\Delta = \Delta'$, $I = I'$, and:

- *(iii) for all $x \in \Delta$, $k_{\mathcal{M}'}(x) \leq k_{\mathcal{M}}(x)$;*
- *(iv) for some $y \in \Delta$, $k_{\mathcal{M}'}(y) < k_{\mathcal{M}}(y)$*

We call $\mathcal{M}$ a minimal enriched model *of K if it is a model of K and there is no $\mathcal{M}'$ model of K such that $\mathcal{M}' <_{Enriched} \mathcal{M}$.*

K minimally entails a query F if F holds in all minimal $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ models of K. We write $K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_{E\,min}} F$. We have developed the semantics above in order to overcome a weakness of rational closure, namely its all-or-nothing character. In order to show that the semantics hits the point, we show here that the semantics is stronger than the one corresponding to rational closure. Furthermore, Example 3 below shows that indeed we have strengthened rational closure by making it possible to separately reason on the different properties. Since the semantic characterization of rational closure is given in terms of rational *canonical* models, here we restrict our attention to enriched rational models which are *canonical*.

Definition 11 (Minimal canonical enriched models of K) *An $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ enriched model $\mathcal{M}$ is a minimal canonical enriched model of K if it satisfies K, it is minimal (with respect to Definition 10) and it is canonical: for all the sets of concepts $\{C_1, C_2, \ldots, C_n\}$ s.t. $K \not\models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E} C_1 \sqcap C_2 \sqcap \cdots \sqcap C_n \sqsubseteq \bot$, there exists (at least) a domain element x such that $x \in (C_1 \sqcap C_2 \sqcap \cdots \sqcap C_n)^I$.*

We call $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E + min - canonical$ the semantics obtained by restricting attention to minimal canonical enriched models. In the following we will write:
$K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E + min - canonical} C \sqsubseteq D$ to mean that $C \sqsubseteq D$ holds in all minimal canonical enriched models of K. The following example shows that this semantics allows us to correctly deal with the wanted inferences of the Introduction, as (**). The fact that the semantics $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E + min - canonical$ is a genuine strengthening of the semantics corresponding to rational closure is formally shown in Theorem 3 below.

Example 3 *Consider any minimal canonical model $\mathcal{M}^*$ of the same K used in Example 1.It can be easily verified that in $\mathcal{M}^*$ there is a domain element y which is a penguin that does not fly and has a nice feather ($y \in Penguin^I, y \in Bird^I, y \in HasNiceFeather^I$). First, it can be verified that $y \in min_<(Penguin^I)$ (by Definition 7, and since by minimality of $<_{Fly}$ and*

$<_{HasNiceFeather}$, $y \in min_{<_{Fly}}(Penguin^I)$ and $y \in min_{<_{HasNiceFeather}}(Penguin^I)$). Furthermore, for all penguin z that has not a nice feather, $y < z$ (again by Definition 7, and since by minimality of $<_{Fly}$ and $<_{HasNiceFeather}$, $y <_{HasNiceFeather} z$). From this, in all minimal canonical $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ models of K it holds that $\mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$, i.e., $K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E+min-canonical} \mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$, which was the wanted inference (**) of the Introduction.

The following theorem is the important technical result of the paper:

Theorem 3 The minimal models semantics $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E + min - canonical$ is stronger than the semantics for rational closure. Let ($K = TBox, ABox$). If $C \sqsubseteq D \in \overline{TBox}$ then $K \models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E+min-canonical} C \sqsubseteq D$.

Proof.(Sketch) By contraposition suppose that $K \not\models_{\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E+min-canonical} C \sqsubseteq D$. Then there is a minimal canonical enriched $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ model $\mathcal{M} = \langle \Delta, <_{A_1}, \ldots, <_{A_n}, <, I \rangle$ of K and an $y \in C^I$ such that $y \notin D^I$. All consistent sets of concepts consistent with K w.r.t. $\mathcal{ALC}^{\mathbf{R}}\mathbf{T}_E$ are also consistent with K with respect to $\mathcal{ALC} + \mathbf{T}_{\mathsf{R}}$, and viceversa (by Theorem 2).By definition of *canonical*, there is also a canonical $\mathcal{ALC} + \mathbf{T}_{\mathsf{R}}$ model of K $\mathcal{M}_{RC} = \langle \Delta, <_R C, I \rangle$ be this model. If C does not contain the $\mathbf{T}$ operator, we are done: in $\mathcal{M}_{RC}$, as in $\mathcal{M}$, there is $y \in C^I$ such that $y \notin D^I$, hence $C \sqsubseteq D$ does not hold in $\mathcal{M}_{RC}$, and $C \sqsubseteq D \notin \overline{TBox}$. If $\mathbf{T}$ occurs in C, and $C = \mathbf{T}(C')$, we still need to show that also in $\mathcal{M}_{RC}$, as in $\mathcal{M}$, $y \in min_{<_{RC}}(C'^I)$. We prove this by showing that for all $x, y \in \Delta$ if $x <_{RC} y$ in $\mathcal{M}_{RC}$, then also $x < y$ in $\mathcal{M}$. The proof is by induction on $k_{\mathcal{M}_{RC}}(x)$.

(a): let $k_{\mathcal{M}_{RC}}(x) = 0$ and $k_{\mathcal{M}_{RC}}(y) > 0$. Since x does not violate any inclusion, also in $\mathcal{M}$ (by minimality of $\mathcal{M}$) for all preference relations $<_{A_j}$ $k_{{A_j}_{\mathcal{M}}}(x) = 0$, and also $k_{\mathcal{M}}(x) = 0$. This cannot hold for y, for which $k_{\mathcal{M}}(y) > 0$ (otherwise $\mathcal{M}$ would violate K, against the hypothesis). Hence $x < y$ in $\mathcal{M}$.

(b): let $k_{\mathcal{M}_{RC}}(x) = i < k_{\mathcal{M}_{RC}}(y)$, i.e. $x <_{RC} y$. As $x <_{RC} y$ in $\mathcal{M}_{RC}$ and the rank of x in $\mathcal{M}_{RC}$ is i, there must be a $\mathbf{T}(B_i) \sqsubseteq A_i \in E_i - E_{i+1}$ such that $x \in (\neg B_i \sqcup A_i)^I$ whereas $y \in (B_i \sqcap \neg A_i)^I$ in $\mathcal{M}_{RC}$. Before we proceed let us notice that by definition of E_i, as well as by what stated just above on the relation between rank of a concept and $k_{\mathcal{M}_{RC}}$, $k_{\mathcal{M}_{RC}}(B_i) = k_{\mathcal{M}_{RC}}(x)$. We will use this fact below. We show that, for any inclusion $\mathbf{T}(B_l) \sqsubseteq A_l \in K$ that is violated by x, it holds that $k_{\mathcal{M}}(B_l) < k_{\mathcal{M}}(B_i)$, so that, by (b), $x < y$.

Let $\mathbf{T}(B_l) \sqsubseteq A_l \in K$ violated by x, i.e. such that $x \in (B_l \sqcap \neg A_l)^I$. Since $\mathcal{M}_{RC}$ satisfies K, there must be $x' <_{RC} x$ in $\mathcal{M}_{RC}$ with $x' \in (B_l)^I$. As $k_{\mathcal{M}_{RC}}(x') < i$, by inductive hypothesis, $x' < x$ in $\mathcal{M}$. As $x' \in {B_l}^I$, $k_{\mathcal{M}}(B_l) \leq k_{\mathcal{M}}(x')$. Since it can be shown that $k_{\mathcal{M}}(x') < k_{\mathcal{M}}(B_i)$, $k_{\mathcal{M}}(B_l) < k_{\mathcal{M}}(B_i)$, and by condition (b), it holds that $x < y$ in $\mathcal{M}$.

With these facts, since $y \in min_{<}(C'^I)$ holds in $\mathcal{M}$, also $y \in min_{<_{RC}}(C'^I)$ in $\mathcal{M}_{RC}$, hence $\mathbf{T}(C') \sqsubseteq D$ does not hold in $\mathcal{M}_{RC}$, and $C \sqsubseteq D = \mathbf{T}(C') \sqsubseteq D \notin \overline{TBox}$.

The theorem follows by contraposition.

Conclusions and Related Works

A lot of work has been done in order to extend the basic formalism of Description Logics (DLs) with nonmonotonic reasoning features (Straccia 1993; Baader and Hollunder 1995; Donini, Nardi, and Rosati 2002; Eiter et al. 2004; Giordano et al. 2007; 2013a; Ke and Sattler 2008; Britz, Heidema, and Meyer 2008; Bonatti, Lutz, and Wolter 2009; Casini and Straccia 2010; Motik and Rosati 2010; Krisnadhi, Sengupta, and Hitzler 2011; Knorr, Hitzler, and Maier 2012; Casini et al. 2013). The purpose of these extensions is to allow reasoning about *prototypical properties* of individuals or classes of individuals.

The interest of rational closure for DLs is that it provides a significant and reasonable skeptical nonmonotonic inference mechanism, while keeping the same complexity as the underlying logic. The first notion of rational closure for DLs was defined by Casini and Straccia (Casini and Straccia 2010). Their rational closure construction for $\mathcal{ALC}$ directly uses entailment in $\mathcal{ALC}$ over a materialization of the KB. A variant of this notion of rational closure has been studied in (Casini et al. 2013), and a semantic characterization for it has been proposed. In (Giordano et al. 2013b; 2015) a notion of rational closure for the logic $\mathcal{ALC}$ has been proposed, building on the notion of rational closure proposed by Lehmann and Magidor (Lehmann and Magidor 1992), together with a minimal model semantics characterization.

It is well known that rational closure has some weaknesses that accompany its well-known qualities, both in the context of propositional logic and in the context of Description Logics. Among the weaknesses is the fact that one cannot separately reason property by property, so that, if a subclass of C is exceptional for a given aspect, it is exceptional "tout court" and does not inherit any of the typical properties of C. Among the strengths of rational closure there is its computational lightness, which is crucial in Description Logics. To overcome the limitations of rational closure, in (Casini and Straccia 2011; 2013) an approach is introduced based on the combination of rational closure and *Defeasible Inheritance Networks*, while in (Casini and Straccia 2012) a lexicographic closure is proposed, and in (Casini et al. 2014) relevant closure, a syntactic stronger version of rational closure.To address the mentioned weakness of rational closure, in this paper we have proposed a finer grained semantics of the semantics for rational closure proposed in (Giordano et al. 2015), where models are equipped with several preference relations. In such a semantics it is possible to relativize the notion of typicality, whence to reason about typical properties independently from each other. We are currently working at the formulation of a syntactic characterization of the semantics which will be a strengthening of rational closure. As the semantics we have proposed provides a strengthening of rational closure, a natural question arises whether this semantics is equivalent to the lexicographic closure proposed in (Lehmann 1995). In particular, lexicographic closure construction for the description logic $\mathcal{ALC}$ has been defined in (Casini and

Straccia 2012). Concerning our Example 3 above, our minimal model semantics gives the same results as lexicographic closure, since $\mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$ can be derived from the lexicographic closure of the TBox and $\mathbf{T}(Penguin) \sqsubseteq HasNiceFeather$ holds in all the minimal canonical enriched models of TBox. However, a general relation needs to be established.

An approach related to our approach is given in (Gil 2014), where it is proposed an extension of $\mathcal{ALC} + \mathbf{T}$ with several typicality operators, each corresponding to a preference relation. This approach is related to ours although different: the language in (Gil 2014) allows for several typicality operators whereas we only have a single typicality operator. The focus of (Gil 2014) is indeed different from ours, as it does not deal with rational closure, whereas this is the main contribution of our paper.

Acknowledgement: This research is partially supported by INDAM-GNCS Project 2016 ”Ragionamento Defeasible nelle Logiche Descrittive”.

References

Baader, F., and Hollunder, B. 1995. Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic. *Journal of Automated Reasoning (JAR)* 15(1):41–68.

Bonatti, P. A.; Lutz, C.; and Wolter, F. 2009. The Complexity of Circumscription in DLs. *Journal of Artificial Intelligence Research (JAIR)* 35:717–773.

Britz, K.; Heidema, J.; and Meyer, T. 2008. Semantic preferential subsumption. In Brewka, G., and Lang, J., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the 11th International Conference (KR 2008)*, 476–484. Sidney, Australia: AAAI Press.

Casini, G., and Straccia, U. 2010. Rational Closure for Defeasible Description Logics. In Janhunen, T., and Niemelä, I., eds., *Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA 2010)*, volume 6341 of *Lecture Notes in Artificial Intelligence*, 77–90. Helsinki, Finland: Springer.

Casini, G., and Straccia, U. 2011. Defeasible Inheritance-Based Description Logics. In Walsh, T., ed., *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, 813–818. Barcelona, Spain: Morgan Kaufmann.

Casini, G., and Straccia, U. 2012. Lexicographic Closure for Defeasible Description Logics. In *Proc. of Australasian Ontology Workshop, vol.969*, 28–39.

Casini, G., and Straccia, U. 2013. Defeasible inheritance-based description logics. *Journal of Artificial Intelligence Research (JAIR)* 48:415–473.

Casini, G.; Meyer, T.; Varzinczak, I. J.; ; and Moodley, K. 2013. Nonmonotonic Reasoning in Description Logics: Rational Closure for the ABox. In *DL 2013, 26th International Workshop on Description Logics*, volume 1014 of *CEUR Workshop Proceedings*, 600–615. CEUR-WS.org.

Casini, G.; Meyer, T.; Moodley, K.; and Nortje, R. 2014. Relevant closure: A new form of defeasible reasoning for description logics. In *JELIA 2014*, 92–106.

Donini, F. M.; Nardi, D.; and Rosati, R. 2002. Description logics of minimal knowledge and negation as failure. *ACM Transactions on Computational Logic (ToCL)* 3(2):177–225.

Eiter, T.; Lukasiewicz, T.; Schindlauer, R.; and Tompits, H. 2004. Combining Answer Set Programming with Description Logics for the Semantic Web. In Dubois, D.; Welty, C.; and Williams, M., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the 9th International Conference (KR 2004)*, 141–151. Whistler, Canada: AAAI Press.

Gil, O. F. 2014. On the non-monotonic description logic alc+t$_{\text{min}}$. *CoRR* abs/1404.6566.

Giordano, L.; Gliozzi, V.; Olivetti, N.; and Pozzato, G. L. 2007. Preferential Description Logics. In Dershowitz, N., and Voronkov, A., eds., *Proceedings of LPAR 2007 (14th Conference on Logic for Programming, Artificial Intelligence, and Reasoning)*, volume 4790 of *LNAI*, 257–272. Yerevan, Armenia: Springer-Verlag.

Giordano, L.; Gliozzi, V.; Olivetti, N.; and Pozzato, G. L. 2009. ALC+T: a preferential extension of Description Logics. *Fundamenta Informaticae* 96:1–32.

Giordano, L.; Gliozzi, V.; Olivetti, N.; and Pozzato, G. L. 2013a. A NonMonotonic Description Logic for Reasoning About Typicality. *Artificial Intelligence* 195:165–202.

Giordano, L.; Gliozzi, V.; Olivetti, N.; and Pozzato, G. L. 2013b. Minimal Model Semantics and Rational Closure in Description Logics . In *26th International Workshop on Description Logics (DL 2013)*, volume 1014, 168 – 180.

Giordano, L.; Gliozzi, V.; Olivetti, N.; and Pozzato, G. L. 2015. Semantic characterization of rational closure: From propositional logic to description logics. *Artificial Intelligence* 226:1–33.

Ke, P., and Sattler, U. 2008. Next Steps for Description Logics of Minimal Knowledge and Negation as Failure. In Baader, F.; Lutz, C.; and Motik, B., eds., *Proceedings of Description Logics*, volume 353 of *CEUR Workshop Proceedings*. Dresden, Germany: CEUR-WS.org.

Knorr, M.; Hitzler, P.; and Maier, F. 2012. Reconciling owl and non-monotonic rules for the semantic web. In *ECAI 2012*, 474479.

Kraus, S.; Lehmann, D.; and Magidor, M. 1990. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence* 44(1-2):167–207.

Krisnadhi, A. A.; Sengupta, K.; and Hitzler, P. 2011. Local closed world semantics: Keep it simple, stupid! In *Proceedings of Description Logics*, volume 745 of *CEUR Workshop Proceedings*.

Lehmann, D., and Magidor, M. 1992. What does a conditional knowledge base entail? *Artificial Intelligence* 55(1):1–60.

Lehmann, D. J. 1995. Another perspective on default reasoning. *Ann. Math. Artif. Intell.* 15(1):61–82.

Motik, B., and Rosati, R. 2010. Reconciling Description Logics and rules. *Journal of the ACM* 57(5).

Straccia, U. 1993. Default inheritance reasoning in hybrid kl-one-style logics. In Bajcsy, R., ed., *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI 1993)*, 676–681. Chambéry, France: Morgan Kaufmann.

Distributing Knowledge into Simple Bases

Adrian Haret and **Jean-Guy Mailly** and **Stefan Woltran**
Institute of Information Systems
TU Wien, Austria
{haret,jmailly,woltran}@dbai.tuwien.ac.at

Abstract

Understanding the behavior of belief change operators for fragments of classical logic has received increasing interest over the last years. Results in this direction are mainly concerned with adapting representation theorems. However, fragment-driven belief change also leads to novel research questions. In this paper we propose the concept of belief distribution, which can be understood as the reverse task of merging. More specifically, we are interested in the following question: given an arbitrary knowledge base K and some merging operator Δ, can we find a profile E and a constraint μ, both from a given fragment of classical logic, such that $\Delta_\mu(E)$ yields a result equivalent to K? In other words, we are interested in seeing if K can be distributed into knowledge bases of simpler structure, such that the task of merging allows for a reconstruction of the original knowledge. Our initial results show that merging based on drastic distance allows for an easy distribution of knowledge, while the power of distribution for operators based on Hamming distance relies heavily on the fragment of choice.

Introduction

Belief change and belief merging have been topics of interest in Artificial Intelligence for three decades (Alchourrón, Gärdenfors, and Makinson 1985; Katsuno and Mendelzon 1991; Konieczny and Pino Pérez 2002). However, the restriction of such operators to specific fragments of propositional logic has received increasing attention only in the last years (Delgrande et al. 2013; Creignou et al. 2014a; 2014b; Zhuang and Pagnucco 2012; Zhuang, Pagnucco, and Zhang 2013; Zhuang and Pagnucco 2014; Delgrande and Peppas 2015; Haret, Rümmele, and Woltran 2015). Mostly, the question tackled in these works is "How should rationality postulates and change operators be adapted to ensure that the result of belief change belongs to a given fragment?". Surprisingly, the question concerning the extent to which the result of a belief change operation can deviate from the fragment under consideration has been neglected so far. In order to tackle this question, we focus here on a certain form of reverse merging. The question is, given an *arbitrary* knowledge base K and some IC-merging (i.e. merging with integrity constraint, see (Konieczny and Pino Pérez 2002)), operator Δ can we find a profile E, *i.e.* a tuple of knowledge bases, and a constraint μ, both from a given *fragment* of classical logic, such that $\Delta_\mu(E)$ yields a result equivalent to K? In other words, we are interested in seeing if K can be distributed into knowledge bases of simpler structure, such that the task of merging allows for a reconstruction of the original knowledge. We call this operation *knowledge distribution*.

Studying the concept of knowledge distribution can be motivated from different points of view. First, consider a scenario where the storage devices have limited expressibility, for instance, databases or logic programs. Our analysis will show which merging operators are required to reconstruct arbitrary knowledge stored in such a set of limited devices. Second, distribution can also be understood as a tool to hide information; only users who know the used merging operator (which thus acts as an encryption key) are able to faithfully retrieve the distributed knowledge. Given the high complexity of belief change (even for revision in "simple" fragments like *Horn* and *2CNF* (Eiter and Gottlob 1992; Liberatore and Schaerf 2001; Creignou, Pichler, and Woltran 2013)), brute-force attack to guess the merging operator is unthinkable. Finally, from the theoretical perspective our results shed light on the power of different merging operators when applied to profiles from certain fragments. In particular, our results show that merging *1CNF* formulas via the Hamming-distance based operator $\Delta^{H,\Sigma}$ does not need additional care, since the result is guaranteed to stay in the fragment.

Related Work. Previous work on merging in fragments of propositional logic proposed an adaptation of existing belief merging operators to ensure that the result of merging belongs to a given fragment (Creignou et al. 2014b), or modified the rationality postulates in order to function in the *Horn* fragment (Haret, Rümmele, and Woltran 2015). Our approach is different, since we do not require that the result of merging stays in a given fragment. On the contrary, we want to decompose arbitrary bases into a fragment-profile. Recent work by Liberatore has also addressed a form of meta-reasoning over belief change operators. In (Liberatore 2015a), the input is a profile of knowledge bases with the expected result of merging R, and the aim is to determine the reliability of the bases (for instance, represented by weights) which allow the obtaining of R. In another paper, Liberatore (2015b) identifies, given a sequence of belief revisions and their results, the initial pre-order which characterizes the revision operator. Finally, even if our approach may seem related to *Knowledge Compilation* (KC) (Darwiche and Marquis 2002; Fargier and Marquis 2014; Marquis 2015), both methods are in fact conceptually different. KC aims at modifying a knowledge base K into a knowledge base K' such that the most important queries for a given application (consistency checking, clausal entailment, model counting, . . .) are simpler to solve with K'. Here, we are interested in the extent to which it is possible to *equivalently represent* an arbitrary knowledge base by simpler fragments when using merging as a recovery operation.

Main Contributions. We formally introduce the concept of knowledge distributability, as well as a restricted version of it where the profile is limited to a single knowledge base (simplifiability). We show that for drastic distance arbitrary knowledge can be distributed into bases restricted to mostly any kind of fragment, while simplifiability is limited to trivial cases. On the other hand, for Hamming-distance based merging the picture is more opaque. We show that for *1CNF*, distributability w.r.t. $\Delta^{H,\Sigma}$ is limited to trivial cases, while slightly more can be done with $\Delta^{H,GMin}$ and $\Delta^{H,GMax}$. For *2CNF* we show that arbitrary knowledge can be distributed and even be simplified. Finally, we discuss the *Horn* fragment for which the results for $\Delta^{H,\Sigma}$, $\Delta^{H,GMin}$ and $\Delta^{H,GMax}$ are situated in between the two former fragments.

Background

Fragments of Propositional Logic. We consider $\mathcal{L}$ as the language of propositional logic over some fixed alphabet $\mathcal{U}$ of propositional atoms. We use standard connectives $\vee$, $\wedge$, $\neg$, and constants $\top$, $\bot$. A clause is a disjunction of literals. A clause is called *Horn* if at most one of its literals is positive. An interpretation is a set of atoms (those set to true). The set of all interpretations is $2^{\mathcal{U}}$. Models of a formula φ are denoted by $Mod(\varphi)$. A knowledge base (KB) is a finite set of formulas and we identify models of a KB K via $Mod(K) = \bigcap_{\varphi \in K} Mod(\varphi)$. A profile is a finite non-empty tuple of KBs. Two formulae φ_1, φ_2 (resp. KBs K_1, K_2) are equivalent, denoted $\varphi_1 \equiv \varphi_2$ (resp. $K_1 \equiv K_2$), when they have the same set of models.

We use a rather general and abstract notion of fragments.

Definition 1. A mapping $Cl : 2^{2^{\mathcal{U}}} \longrightarrow 2^{2^{\mathcal{U}}}$ is called *closure-operator* if it satisfies the following for any $\mathcal{M}, \mathcal{N} \subseteq 2^{\mathcal{U}}$:

- If $\mathcal{M} \subseteq \mathcal{N}$, then $Cl(\mathcal{M}) \subseteq Cl(\mathcal{N})$
- If $|\mathcal{M}| = 1$, then $Cl(\mathcal{M}) = \mathcal{M}$
- $Cl(\emptyset) = \emptyset$.

Definition 2. $\mathcal{L}' \subseteq \mathcal{L}$ is called a *fragment* if it is closed under conjunction (i.e., $\varphi \wedge \psi \in \mathcal{L}'$ for any $\varphi, \psi \in \mathcal{L}'$), and there exists an associated closure-operator Cl such that (1) for all $\psi \in \mathcal{L}'$, $Mod(\psi) = Cl(Mod(\psi))$ and (2) for all $\mathcal{M} \subseteq 2^{\mathcal{U}}$ there is a $\psi \in \mathcal{L}'$ with $Mod(\psi) = Cl(\mathcal{M})$. We often denote the closure-operator Cl associated to a fragment $\mathcal{L}'$ as $Cl_{\mathcal{L}'}$.

Definition 3. For a fragment $\mathcal{L}'$, we call a finite set $K \subseteq \mathcal{L}'$ an $\mathcal{L}'$-knowledge base. An $\mathcal{L}'$-profile is a profile over $\mathcal{L}'$-knowledge bases. A KB $K' \subseteq \mathcal{L}$ is called *$\mathcal{L}'$-expressible* if there exists an $\mathcal{L}'$-KB K, such that $K' \equiv K$.

Many well known fragments of propositional logic are indeed captured by our notion. For the Horn-fragment $\mathcal{L}_{Horn}$, i.e. the set of all conjunctions of Horn clauses over $\mathcal{U}$, take the operator $Cl_{\mathcal{L}_{Horn}}$ defined as the fixed point of the function

$$Cl^1_{\mathcal{L}_{Horn}}(\mathcal{M}) = \{\omega_1 \cap \omega_2 \mid \omega_1, \omega_2 \in \mathcal{M}\}.$$

The fragment $\mathcal{L}_{2CNF}$ which is restricted to formulas over clauses of length at most 2 is linked to the operator $Cl_{\mathcal{L}_{2CNF}}$ defined as the fixed point of the function $Cl^1_{\mathcal{L}_{2CNF}}$ given by

$$Cl^1_{\mathcal{L}_{2CNF}}(\mathcal{M}) = \{\mathrm{maj}_3(\omega_1, \omega_2, \omega_3) \mid \omega_1, \omega_2, \omega_3 \in \mathcal{M}\}.$$

Here, we use the ternary majority function $\mathrm{maj}_3(\omega_1, \omega_2, \omega_3)$ which yields an interpretation containing those atoms which are true in at least two out of $\omega_1, \omega_2, \omega_3$. Finally, we are also interested in the $\mathcal{L}_{1CNF}$ fragment which is just composed of conjunctions of literals; its associated operator $Cl_{\mathcal{L}_{1CNF}}$ is defined as the fixed point of the function

$$\begin{aligned} Cl^1_{\mathcal{L}_{1CNF}}(\mathcal{M}) \quad = \quad & \{\omega_1 \cap \omega_2, \omega_1 \cup \omega_2 \mid \omega_1, \omega_2 \in \mathcal{M}\} \cup \\ & \{\omega_3 \mid \omega_1 \subseteq \omega_3 \subseteq \omega_2; \omega_1, \omega_2 \in \mathcal{M}\}. \end{aligned}$$

Note that full classical logic is given via the identity closure operator $Cl_{\mathcal{L}}(\mathcal{M}) = \mathcal{M}$.

Merging Operators. We focus on *IC-merging*, where a profile is mapped into a KB, such that the result satisfies some integrity constraint. Postulates for IC-merging have been stated in (Konieczny and Pino Pérez 2002). We recall a specific family of IC-merging operators, based on distances between interpretations, see also (Konieczny, Lang, and Marquis 2004).

Definition 4. A distance between interpretations is a mapping d from two interpretations to a non-negative real number, such that for all $\omega_1, \omega_2, \omega_3 \subseteq \mathcal{U}$, (1) $d(\omega_1, \omega_2) = 0$ iff $\omega_1 = \omega_2$; (2) $d(\omega_1, \omega_2) = d(\omega_2, \omega_1)$; and (3) $d(\omega_1, \omega_2) + d(\omega_2, \omega_3) \geq d(\omega_1, \omega_3)$. We will use two specific distances:

drastic distance $D(\omega_1, \omega_2) = 1$ if $\omega_1 = \omega_2$, 0 otherwise;

Hamming distance $H(\omega_1, \omega_2) = |(\omega_1 \setminus \omega_2) \cup (\omega_2 \setminus \omega_1)|$.

We overload the previous notations to define the distance between an interpretation ω and a KB K: if d is a distance between interpretations, then

$$d(\omega, K) = \min_{\omega' \in Mod(K)} d(\omega, \omega').$$

Next, an aggregation function must be used to evaluate the distance between an interpretation and a profile.

Definition 5. An aggregation function $\otimes$ associates a non-negative number to every finite tuple of non-negative numbers, such that:

1. If $x \leq y$, then $\otimes(x_1, \dots, x, \dots, x_n) \leq \otimes(x_1, \dots, y, \dots, x_n)$;
2. $\otimes(x_1, \dots, x_n) = 0$ iff $x_1 = \cdots = x_n = 0$;
3. For every non-negative number x, $\otimes(x) = x$.

As aggregation functions, we will consider the sum Σ, and GMax and GMin[1], defined as follows. Given a profile $(K_1, \dots, K_n)$, let $V_\omega = (d_1^\omega, \dots, d_n^\omega)$ be the vector of distances s.t. $d_i^\omega = d(\omega, K_i)$. $\mathit{GMax}(d_1^\omega, \dots, d_n^\omega)$ (resp. $\mathit{GMin}(d_1^\omega, \dots, d_n^\omega)$) is defined by ordering V_ω in decreasing (resp. increasing) order. Given two interpretations ω_1, ω_2, $\mathit{GMax}(d_1^{\omega_1}, \dots, d_n^{\omega_1}) \leq \mathit{GMax}(d_1^{\omega_2}, \dots, d_n^{\omega_2})$ (resp. $\mathit{GMin}(d_1^{\omega_1}, \dots, d_n^{\omega_1}) \leq \mathit{GMin}(d_1^{\omega_2}, \dots, d_n^{\omega_2})$) is defined by comparing them w.r.t. the lexicographic ordering.

Finally, let d be a distance, ω an interpretation and $E = (K_1, \dots, K_n)$ a profile. Then,

$$d^{\otimes}(\omega, E) = \otimes(d(\omega, K_1), \dots, d(\omega, K_n)).$$

If there is no ambiguity about the aggregation function $\otimes$, we write $d(\omega, E)$ instead of $d^{\otimes}(\omega, E)$.

Definition 6. For any distance d between interpretations, and any aggregation function $\otimes$, the merging operator $\Delta^{d,\otimes}$ is a mapping from a profile E and a formula μ to a KB, such that

$$\mathit{Mod}(\Delta_\mu^{d,\otimes}(E)) = \min(\mathit{Mod}(\mu), \leq_E^{d,\otimes}),$$

with $\omega_1 \leq_E^{d,\otimes} \omega_2$ iff $d^{\otimes}(\omega_1, E) \leq d^{\otimes}(\omega_2, E)$.

When we consider a profile containing a single knowledge base K, all aggregation functions are equivalent; we write $\Delta_\mu^d(K)$ instead of $\Delta_\mu^{d,\otimes}((K))$ for readability. For drastic distance, GMin, GMax, and Σ are equivalent for arbitrary profiles. Thus, whenever we show results for $\Delta^{D,\Sigma}$, these carry over to $\Delta^{D,\mathit{GMin}}$ and $\Delta^{D,\mathit{GMax}}$.

Main Concepts and General Results

We now give the central definition for a knowledge base being distributable into a profile from a certain fragment with respect to a given merging operator.

Definition 7. Let Δ be a merging operator, $K \subseteq \mathcal{L}$ be an arbitrary KB, and $\mathcal{L}'$ be a fragment. K is called *$\mathcal{L}'$-distributable* w.r.t. Δ if there exists an $\mathcal{L}'$-profile E and a formula $\mu \in \mathcal{L}'$, such that $\Delta_\mu(E) \equiv K$.

Example 1. Let $\mathcal{U} = \{a, b\}$ and consider $K = \{a \vee b\}$ which we want to check for $\mathcal{L}_{\mathit{Horn}}$-distributability w.r.t. operator $\Delta^{H,\Sigma}$. We have $\mathit{Mod}(K) = \{\{a\}, \{b\}, \{a, b\}\}$, thus K is not $\mathcal{L}_{\mathit{Horn}}$-expressible (note that $\mathit{Cl}_{\mathcal{L}_{\mathit{Horn}}}(\mathit{Mod}(K)) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\} \neq \mathit{Mod}(K)$), otherwise K would be distributable in a simple way (see Proposition 1 below).

Take the $\mathcal{L}_{\mathit{Horn}}$-profile $E = (K_1, K_2)$ with $K_1 = \{a \wedge b\}$, $K_2 = \{\neg a \vee \neg b\}$, together with the empty constraint $\mu = a \vee \neg a$. We have $\mathit{Mod}(K_1) = \{\{a, b\}\}$, $\mathit{Mod}(K_2) = \{\{a\}, \{b\}, \emptyset\}$. In the following matrix, each line corresponds to the distance between a model of μ and a KB from the profile E (columns K_1 and K_2), or between a model of μ and the profile using the sum-aggregation over the distances to the single KBs (column Σ).

	K_1	K_2	Σ
$\{a,b\}$	0	1	1
$\{a\}$	1	0	1
$\{b\}$	1	0	1
$\emptyset$	2	0	2

We observe that $\mathit{Mod}(\Delta_\mu^{H,\Sigma}(E)) = \{\{a\}, \{b\}, \{a, b\}\}$, thus $\Delta_\mu^{H,\Sigma}(E) \equiv K$ as desired. It is easily checked that also other aggregations work: $\Delta_\mu^{H,\mathit{GMax}}(E) \equiv \Delta_\mu^{H,\mathit{GMin}}(E) \equiv K$. $\diamond$

Next, we recall that IC-merging of a single KB yields revision. Thus, the concept we introduce next is also of interest, as it represents a certain form of reverse revision.

Definition 8. Let Δ be a merging operator, $K \subseteq \mathcal{L}$ an arbitrary KB, and $\mathcal{L}'$ a fragment. K is called *$\mathcal{L}'$-simplifiable* w.r.t. Δ if there exists an $\mathcal{L}'$-KB K' and $\mu \in \mathcal{L}'$, such that $\Delta_\mu(K') \equiv K$.

As we will see later, the KB K from Example 1 cannot be $\mathcal{L}_{\mathit{Horn}}$-simplified w.r.t. Δ^H; in other words, we need here at least two KBs to "express" K. However, it is rather straightforward that any $\mathcal{L}'$-expressible KB can be $\mathcal{L}'$-simplified.

Proposition 1. *For every fragment $\mathcal{L}'$ and every KB K, it holds that K is $\mathcal{L}'$-simplifiable (and thus also $\mathcal{L}'$-distributable) w.r.t. Δ, whenever K is $\mathcal{L}'$-expressible.*

Proof. Let K' be an $\mathcal{L}'$-KB equivalent to K, and let $\mu = (\bigwedge_{\varphi \in K'} \varphi)$. Thus, $\mu \in \mathcal{L}'$ by definition of fragments and it is easily verified that $\Delta_\mu(K') \equiv K$. □

Next, we show that in order to determine whether a KB K is $\mathcal{L}'$-distributable, it is sufficient to consider constraints μ such that $\mathit{Mod}(\mu) = \mathit{Cl}_{\mathcal{L}'}(\mathit{Mod}(K))$.

Proposition 2. *Let $K \in \mathcal{L}$ be a KB, $\mathcal{L}'$ be a fragment, E an $\mathcal{L}'$-profile and $\mu \in \mathcal{L}'$. Then $\Delta_\mu(E) \equiv K$ implies $\Delta_{\mu'}(E) \equiv K$ for any μ' such that $\mathit{Mod}(\mu') = \mathit{Cl}_{\mathcal{L}'}(\mathit{Mod}(K))$.*

Proof. Let $\Delta = \Delta^{d,\otimes}$. By Definition 6, $\mathit{Mod}(K) = \min(\mathit{Mod}(\mu), \leq_E^{d,\otimes})$, hence $\mathit{Mod}(K) \subseteq \mathit{Mod}(\mu)$. Moreover, μ is $\mathcal{L}'$-closed, so $\mathit{Cl}_{\mathcal{L}'}(\mathit{Mod}(K)) = \mathit{Mod}(\mu') \subseteq \mathit{Mod}(\mu)$. We get $\mathit{Mod}(K) \subseteq \mathit{Mod}(\mu') \subseteq \mathit{Mod}(\mu)$. Thus, $\mathit{Mod}(K) = \min(\mathit{Mod}(\mu'), \leq_E^{d,\otimes})$, i.e. $\Delta_{\mu'}(E) \equiv K$. □

Next, we give two positive results for distributing knowledge in any fragment. The key idea is to use KBs in the profile which have exactly one model (our notion of fragment guarantees existence of such KBs). The first result is independent of the distance notion but requires GMin as the aggregation function. The second result is for drastic distance and thus works for any of the aggregation functions we consider.

[1] GMax and GMin are also known as *leximax* and *leximin* respectively. Stricto sensu, these functions return a vector of numbers, and not a single number. However, GMax (resp. GMin) can be associated with an aggregation function as defined in Definition 5 which yields the same vector ordering than GMax (resp. GMin). We do a slight abuse by using directly GMax and GMin as the names of aggregation functions. See (Konieczny, Lang, and Marquis 2002).

Theorem 3. *Let d be a distance and $\mathcal{L}'$ be a fragment. Then for every KB K, such that for all distinct $\omega_1, \omega_2 \in Mod(K)$, $d(\omega_1, \omega_2) = e$ for some $e > 0$, it holds that K is $\mathcal{L}'$-distributable w.r.t. $\Delta^{d,GMin}$.*

Proof. Build the $\mathcal{L}'$-profile E such that for each $\omega \in Mod(K)$, there is a KB with ω as its only model. Thus all models of K get a $GMin$-vector $(0, e, e, e, e, \ldots)$. All interpretations from $Cl_{\mathcal{L}'}(Mod(K)) \setminus Mod(K)$ get a vector $(f, g, \ldots)$ with $f > 0$. Hence, we have $\min(Mod(\mu), \leq_E^{d,GMin}) = Mod(K)$ using $\mu \in \mathcal{L}'$ with $Mod(\mu) = Cl_{\mathcal{L}'}(Mod(K))$. □

Theorem 4. *For every fragment $\mathcal{L}'$ and every knowledge base K, it holds that K is $\mathcal{L}'$-distributable w.r.t. $\Delta^{D,\oplus}$, for $\oplus \in \{\Sigma, GMin, GMax\}$.*

Proof. Given a fragment $\mathcal{L}'$, we take $E = \{K_\omega \mid \omega \in Mod(K)\}$ where $K_\omega \in \mathcal{L}'$ is a knowledge base with single model ω (such $K_\omega \in \mathcal{L}'$ exists due to our definition of fragments), and let μ be such that $Mod(\mu) = Cl_{\mathcal{L}'}(Mod(K))$; hence also $\mu \in \mathcal{L}'$. Let $\omega' \in Mod(\mu)$ and $n = |Mod(K)|$, we observe that $\Sigma_{K_\omega \in E} H(\omega', K_\omega) = n - 1$ when $\omega' \in Mod(K)$, and n otherwise. Thus, $\Delta_\mu^{D,\Sigma}(E) \equiv K$. The same result holds for $\Delta_\mu^{D,GMax}$ and $\Delta_\mu^{D,GMin}$. □

Concerning simplifiability w.r.t. drastic distance based operators, Proposition 1 cannot be improved.

Theorem 5. *For every fragment $\mathcal{L}'$ and every KB K, K is $\mathcal{L}'$-*simplifiable *w.r.t. Δ^D iff K is $\mathcal{L}'$-expressible.*

Proof. The if-direction is by Proposition 1. For the other direction, suppose K is not $\mathcal{L}'$-expressible. We show that for any $\mathcal{L}'$-KB K', $\Delta_\mu^D(K') \not\equiv K$ with $\mu = Cl_{\mathcal{L}'}(K)$. By Proposition 2 the result then follows. Now suppose there exists an $\mathcal{L}'$-KB K' such that $\Delta_\mu^D(K') \equiv K$. First observe that since K is not $\mathcal{L}'$-expressible, $Mod(\mu) \supset Mod(K)$. Since we are working with drastic distance, in order to promote models of K, we also need them in K', hence $Mod(K') \supseteq Mod(K)$ and since K' is from $\mathcal{L}'$ we have $Mod(K') \supseteq Cl_{\mathcal{L}'}(K) = Mod(\mu)$. Thus there exists $\omega \in Cl_{\mathcal{L}'}(Mod(K)) \setminus Mod(K)$ having distance 0 to K', and thus $\omega \in \Delta_\mu^D(K')$. Since $\omega \notin Mod(K)$, this yields a contradiction to $\Delta_\mu^D(K') \equiv K$. □

Hamming Distance and Specific Fragments

We first consider the simplest fragment under consideration, namely conjunction of literals. As it turns out, (non-trivial) distributability for this fragment w.r.t. $\Delta^{H,\Sigma}$ is not achievable. We then see that more general fragments allow for non-trivial distributions. In particular, we show that every KB is distributable (and even simplifiable) in the *2CNF* case, and we finally give a few observations for $\mathcal{L}_{Horn}$.

The 1CNF Fragment

The following technical result is important to prove the main result in this section.

Lemma 6. *For any $\mathcal{L}_{1CNF}$-profile $E = (K_1, \ldots, K_n)$ and interpretations ω_1, ω_2, it holds that:*

$$H(\omega_1, E) + H(\omega_2, E) = H(\omega_1 \cap \omega_2, E) + H(\omega_1 \cup \omega_2, E).$$

Proof. It suffices to show that for each K_i in profile E, $H(\omega_1, K_i) + H(\omega_2, K_i) = H(\omega_1 \cap \omega_2, K_i) + H(\omega_1 \cup \omega_2, K_i)$. Indeed, summing up these equalities over all $K_i \in E$, we get

$$\Sigma_{K_i \in E} H(\omega_1, K_i) + \Sigma_{K_i \in E} H(\omega_2, K_i) =$$
$$\Sigma_{K_i \in E} H(\omega_1 \cap \omega_2, K_i) + \Sigma_{K_i \in E} H(\omega_1 \cup \omega_2, K_i).$$

Since $H(\omega, E) = \Sigma_{K_i \in E} H(\omega, K_i)$, for any interpretation ω, our conclusion then follows immediately.

Thus, take ω_1', ω_2' to be two interpretations that are closest to ω_1 and ω_2, respectively, among the models of $Mod(K_i)$. In other words, $H(\omega_1, \omega_1') = \min_{\omega \in Mod(K_i)} H(\omega_1, \omega)$ and $H(\omega_2, \omega_2') = \min_{\omega \in Mod(K_i)} H(\omega_2, \omega)$. By induction on the number of propositional atoms in $\mathcal{L}$, we can show that $\omega_1' \cap \omega_2'$ and $\omega_1' \cup \omega_2'$ are closest in $Mod(K_i)$ to $\omega_1 \cap \omega_2$ and $\omega_1 \cup \omega_2$, respectively. Thus, we have that $H(\omega_1, K_i) = H(\omega_1, \omega_1')$, $H(\omega_2, K_i) = H(\omega_2, \omega_2')$, $H(\omega_1 \cap \omega_2, K_i) = H(\omega_1 \cap \omega_2, \omega_1' \cap \omega_2')$, $H(\omega_1 \cup \omega_2, K_i) = H(\omega_1 \cup \omega_2, \omega_1' \cup \omega_2')$, and our problem reduces to showing that $H(\omega_1, \omega_1') + H(\omega_2, \omega_2') = H(\omega_1 \cap \omega_2, \omega_1' \cap \omega_2') + H(\omega_1 \cup \omega_2, \omega_1' \cup \omega_2')$. By using induction on the number of propositional atoms in $\mathcal{L}$ again, we can show that this equality holds. The argument runs as follows: in the base case, when the alphabet consists of just one propositional atom, the equality is shown to be true by checking all the cases. For the inductive step we assume the claim holds for an alphabet of size n and show that it also holds for an alphabet of size $n + 1$. More concretely, we analyze the way in which the Hamming distances between interpretations change when we add a propositional atom to the alphabet. An analysis of all the possible cases shows that the equality holds. □

Next we observe certain patterns of interpretations that indicate whether a KB is $\mathcal{L}_{1CNF}$-expressible or not.

Definition 9. If K is a knowledge base, then a pair of interpretations ω_1 and ω_2 are called *critical with respect to K* if $\omega_1 \not\subseteq \omega_2$ and $\omega_2 \not\subseteq \omega_1$, and one of the following cases holds:

1. $\omega_1, \omega_2 \in Mod(K)$ and $\omega_1 \cap \omega_2, \omega_1 \cup \omega_2 \notin Mod(K)$,
2. $\omega_1, \omega_2, \omega_1 \cap \omega_2 \in Mod(K)$ and $\omega_1 \cup \omega_2 \notin Mod(K)$,
3. $\omega_1, \omega_2, \omega_1 \cup \omega_2 \in Mod(K)$ and $\omega_1 \cap \omega_2 \notin Mod(K)$,
4. $\omega_1 \cap \omega_2, \omega_1 \cup \omega_2 \in Mod(K)$ and $\omega_1, \omega_2 \notin Mod(K)$, or
5. $\omega_1, \omega_1 \cap \omega_2, \omega_1 \cup \omega_2 \in Mod(K)$ and $\omega_2 \notin Mod(K)$.

Lemma 7. *If a KB K is not $\mathcal{L}_{1CNF}$-expressible, then there exist $\omega_1, \omega_2 \in Cl_{\mathcal{L}_{1CNF}}(K)$ being critical with respect to K.*

Proof. The fact that K is not $\mathcal{L}_{1CNF}$-expressible implies that either: (i) K is not closed under intersection or union, or (ii) there are $w_1, w_2, w_3 \in Cl_{\mathcal{L}_{1CNF}}(K)$ such that $w_1 \subseteq w_3 \subseteq w_2$, and $w_1, w_2 \in Mod(K)$, $w_3 \notin Mod(K)$. Case (i) implies that there exist $w_1, w_2 \in Mod(K)$ such that one of Cases 1-3 from Definition 9 holds. If we are in Case (ii), then consider the interpretation $w_4 = (w_2 \setminus w_3) \cup w_1$. Clearly, $w_1 \subseteq w_4 \subseteq w_2$, hence $w_4 \in Cl_{\mathcal{L}_{1CNF}}(K)$. Also,

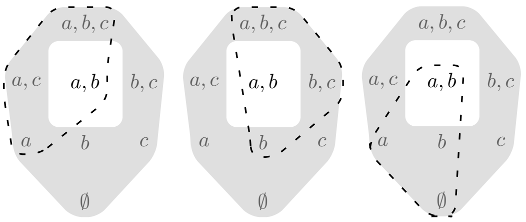
$w_3 \cap w_4 = w_1$ and $w_3 \cup w_4 = w_2$. There are two sub-cases to consider here. If $w_4 \notin Mod(K)$, then we are in Case 4 of Definition 9. If $w_4 \in Mod(K)$, then we are in Case 5 of Definition 9. □

Example 2. Let us consider the KB K such that $Mod(K) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$. K is not *1CNF*-expressible; indeed, $Cl_{1CNF}(Mod(K)) = Mod(K) \cup \{\{a, b\}\}$.

Here, we identify several sets of critical interpretations w.r.t. K. First, $S_1 = \{\{a, c\}, \{a, b\}, \{a\}, \{a, b, c\}\}$ corresponds to the situation described in Case 5 of Definition 9, with $\omega_1 = \{a, c\}$ and $\omega_2 = \{a, b\}$.

The set $S_2 = \{\{b, c\}, \{a, b\}, \{b\}, \{a, b, c\}\}$ also corresponds to Case 5, with $\omega_1 = \{b, c\}$ and $\omega_2 = \{a, b\}$.

We can also consider the set of interpretations $S_3 = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$, which corresponds to Case 2 of Definition 9, with $\omega_1 = \{a\}$ and $\omega_2 = \{b\}$. The models of K and the sets of critical interpretations are represented in Figure 1.



(a) S_1 (b) S_2 (c) S_3

Figure 1: Models of K are in the shaded area; critical interpretations are in the dashed areas.

We can now state the central result of this section.

Theorem 8. *A KB K is $\mathcal{L}_{1CNF}$-distributable with respect to $\Delta^{H,\Sigma}$ if and only if K is $\mathcal{L}_{1CNF}$-expressible.*

Proof. If part. By Proposition 1.

Only if part. Let K be a KB that is not $\mathcal{L}_{1CNF}$-expressible. We will show that it is not $\mathcal{L}_{1CNF}$-distributable w.r.t. $\Delta^{H,\Sigma}$. Suppose, on the contrary, that K is $\mathcal{L}_{1CNF}$-distributable. Then there exists an $\mathcal{L}_{1CNF}$ profile $E = (K_1, \ldots, K_n)$ such that $\Delta^{H,\Sigma}_{\mu}(E) \equiv K$, where $Mod(\mu) = Cl_{\mathcal{L}_{1CNF}}(Mod(K))$ (cf. Proposition 2).

By Lemma 7, there exist interpretations $\omega_1, \omega_2 \in Mod(\mu)$ that are critical with respect to K. By Lemma 6, we have

$$H(\omega_1, E) + H(\omega_2, E) = H(\omega_1 \cap \omega_2, E) + H(\omega_1 \cup \omega_2, E). \quad (1)$$

Let us now do a case analysis depending on the type of critical pair we are dealing with. If we are in Case 1 of Definition 9, then it needs to be the case that $H(\omega_1, E) = H(\omega_2, E) = m$, $H(\omega_1 \cap \omega_2, E) = m + k_1$ and $H(\omega_1 \cup \omega_2, E) = m + k_2$, for some integers $m \geq 0$ and $k_1, k_2 > 0$. Plugging these numbers into Equality (1), we get that $2m = 2m + k_1 + k_2$ and $k_1 + k_2 = 0$. Since $k_1, k_2 > 0$, we have arrived at a contradiction. If we are in Case 2, then it needs to be the case that $H(\omega_1 \cap \omega_2, E) = H(\omega_1 \cup \omega_2, E) = m$, $H(\omega_1, E) = m + k_1$ and $H(\omega_2, E) = m + k_2$, for some integers $m \geq 0$ and $k_1, k_2 > 0$. Plugging these numbers into Equality (1) again, we get a contradiction along the same lines as in Case 1. If we are in Case 3, then it needs to hold that $H(\omega_1, E) = H(\omega_1 \cap \omega_2, E) = H(\omega_1 \cup \omega_2, E) = m$, $H(\omega_2, E) = m + k$, for some integers $m \geq 0$ and $k > 0$. Plugging these numbers into Equality (1) gives us $2m + k = 2m$ and hence $k = 0$. Since $k > 0$, we have arrived at a contradiction. Cases 4 and 5 are entirely similar. □

In other words, for any $\mathcal{L}_{1CNF}$-profile and $\mu \in$ *1CNF*, $\Delta^{H,\Sigma}_{\mu}$ is guaranteed to be $\mathcal{L}_{1CNF}$-expressible as well. As we have already shown in Theorem 3, this is not necessarily the case if we replace Σ by *GMin*. The following example shows how to obtain a similar behavior for *GMax*; we then generalize this idea below.

Example 3. Let $\mathcal{U} = \{a, b\}$ and $K = \{a \vee b, \neg a \vee \neg b\}$. We have $Mod(K) = \{\{a\}, \{b\}\}$. K is not $\mathcal{L}_{1CNF}$-expressible, since $Cl_{\mathcal{L}_{1CNF}}(Mod(K)) = 2^{\mathcal{U}}$. Let K_S be the $\mathcal{L}_{1CNF}$-KB with a single model S for any $S \subseteq \mathcal{U}$ and let us have a look at the following distance matrix for μ with $Mod(\mu) = Cl_{\mathcal{L}_{1CNF}}(Mod(K))$, $E = (K_{\{a\}}, K_{\{b\}})$, and $E' = (K_{\emptyset}, K_{\{a,b\}})$.

	$K_{\emptyset}$	$K_{\{a\}}$	$K_{\{b\}}$	$K_{\{a,b\}}$	$H^{GMin}(E)$	$H^{GMax}(E')$
$\emptyset$	0	1	1	2	$(1, 1)$	$(2, 0)$
$\{a\}$	1	0	2	1	$(0, 2)$	$(1, 1)$
$\{b\}$	1	2	0	1	$(0, 2)$	$(1, 1)$
$\{a, b\}$	2	1	1	0	$(1, 1)$	$(2, 0)$

Recall that the lexicographic order of the involved vectors is $(0, 2) < (1, 1) < (2, 0)$. We thus get that $\Delta^{H,GMin}_{\mu}(E) \equiv K$ (see also Theorem 3), and on the other hand, $\Delta^{H,GMax}_{\mu}(E') \equiv K$. ◇

Theorem 9. *Any KB K such that $Mod(K) = \{\omega, \omega'\}$ is $\mathcal{L}_{1CNF}$-distributable with respect to $\Delta^{H,GMax}$.*

Proof. If K is $\mathcal{L}_{1CNF}$-expressible, then the conclusion follows from Proposition 1. If K is not $\mathcal{L}_{1CNF}$-expressible, then consider the set $Cl_{\mathcal{L}_{1CNF}}(Mod(K)) \setminus Mod(K) = \{\omega_1, \ldots, \omega_n\}$. We define the profile $E = (K_1, \ldots, K_n)$, where $Mod(K_i) = \{\mathcal{U} \setminus \omega_i\}$, for $i \in \{1, \ldots, n\}$. We show that $\Delta^{H,GMax}_{\mu}(E) \equiv K$, where $Mod(\mu) = Cl_{\mathcal{L}_{1CNF}}(Mod(K))$.

First, we have that $H(\omega_i, \mathcal{U} \setminus \omega_i) = |\mathcal{U}|$, which implies that $H^{GMax}(\omega_i, E) = GMax(|\mathcal{U}|, \ldots)$, for any $i \in \{1, \ldots, n\}$. Furthermore, since $H(\omega, \mathcal{U} \setminus \omega_i) < |\mathcal{U}|$ and $H(\omega', \mathcal{U} \setminus \omega_i) < |\mathcal{U}|$, for any $i \in \{1, \ldots, n\}$, it follows that $\omega <^{H,GMax}_{E} \omega_i$ and $\omega' <^{H,GMax}_{E} \omega_i$. Next, we show that $H^{GMax}(\omega, E) = H^{GMax}(\omega', E)$.

Consider the vectors $V = (H(\omega, \omega_1), \ldots, H(\omega, \omega_n))$ and $V' = (H(\omega', \omega_1), \ldots, H(\omega', \omega_n))$. Our claim is that $GMax(V) = GMax(V')$. To see why, notice that the elements in $Cl_{\mathcal{L}_{1CNF}}(Mod(K))$ form a complete subset lattice with $\omega \cup \omega'$ and $\omega \cap \omega'$ as the top and bottom elements, respectively. Let us write $H(\omega, \omega') = m$. This lattice has 2^m elements, and the maximum distance of two elements in it is m. Thus, the vector V is the vector of distances between ω and every other element in this lattice, except itself and ω'. A similar consideration holds for V'. Hence V and V' are vectors of length 2^{m-2} whose elements are $m-1, m-2, \ldots, 1$. We can actually count how many times each number appears in V and V'. The number of interpretations in the lattice that are at distance of 1 from ω (and ω') is $\binom{1}{m}$: thus, $m-1$ appears $\binom{1}{m}$ times in V (and V'). The number of interpretations that are at distance 2 from ω (and ω') is $\binom{2}{m}$, thus $m-2$ appears $\binom{2}{m}$ times in V and V'. We iterate this argument for every distance, up to 1. It is then easy to see that, based on these considerations, V and V' are equal when sorted in descending order. Our conclusion follows from this. □

The 2CNF Fragment

We show that every knowledge base K can be distributed in the fragment $\mathcal{L}_{2CNF}$. Even a single $\mathcal{L}_{2CNF}$ knowledge base is enough to represent K. Before giving the general result, we sketch the idea via an example.

Example 4. Let K be a KB with $Mod(K) = \{\{a,b\},\{b,c,e\},\{a,c,d\}\}$. We observe that K is not $\mathcal{L}_{2CNF}$-expressible since $Cl_{\mathcal{L}_{2CNF}}(Mod(K)) = Mod(K) \cup \{a,b,c\}$. However, we can give an $\mathcal{L}_{2CNF}$-KB K' using three new atoms x, y, z to penalize the undesired interpretation $\{a,b,c\}$ such that $\Delta^H_\mu(K') \equiv K$, with $\mu \in \mathcal{L}_{2CNF}$ of the form $Mod(\mu) = Cl_{\mathcal{L}_{2CNF}}(Mod(K))$. To this end, assume K' with $Mod(K') = \{\omega_1, \omega_2, \omega_3, \omega_4\}$ of the form

$$\begin{aligned}
\omega_1 &= \{a,b,x,y\},\\
\omega_2 &= \{b,c,e,x,z\},\\
\omega_3 &= \{a,c,d,y,z\},\\
\omega_4 &= \{a,b,c,x,y,z\}.
\end{aligned}$$

One can verify that $Cl_{\mathcal{L}_{2CNF}}(K') = Mod(K')$. Thus, K' can be picked from $\mathcal{L}_{2CNF}$. We use μ such that $Mod(\mu) = Cl_{\mathcal{L}_{2CNF}}(Mod(K))$ and get distances

	ω_1	ω_2	ω_3	ω_4	min
$\{a,b\}$	2	5	5	4	2
$\{b,c,e\}$	4	2	6	4	2
$\{a,c,d\}$	4	6	2	4	2
$\{a,b,c\}$	3	4	4	3	3

Here, each line gives the distance between a model of μ and a model of K' (ω_i columns), or between a model of μ and K' (min column). The key observation is that pairs from x, y, z as used in ω_1, ω_2, ω_3 give minimal distances 2 while the remaining interpretation ω_4, which corresponds to the closure of K, contains all three new atoms (since $maj3(\{x,y\},\{x,z\},\{y,z\}) = \{x,y,z\}$). ◇

Theorem 10. *Any KB K is $\mathcal{L}_{2CNF}$-simplifiable w.r.t. Δ^H_μ.*

Proof. We have to show that for any KB K, there exists an $\mathcal{L}_{2CNF}$-KB K' and a formula $\mu \in \mathcal{L}_{2CNF}$ such that $\Delta^H_\mu(K') \equiv K$. If K is $\mathcal{L}_{2CNF}$-expressible, the result is due to Proposition 1. So suppose that K is not $\mathcal{L}_{2CNF}$-expressible and let $Mod(K) = \{\omega_1, \ldots, \omega_n\}$. Consider a set of new atoms $A = \{a_1, \ldots, a_n\}$, and for each $\omega_i \in Mod(K)$, let $\omega'_i = \omega_i \cup A \setminus \{a_i\}$. We define the $\mathcal{L}_{2CNF}$-KB K' and $\mu \in \mathcal{L}_{2CNF}$ such that

$$\begin{aligned}
Mod(K') &= Cl_{\mathcal{L}_{2CNF}}(\{\omega'_i \mid \omega_i \in Mod(K)\})\\
Mod(\mu) &= Cl_{\mathcal{L}_{2CNF}}(Mod(K)).
\end{aligned}$$

Let $\Omega' = \{\omega'_i \mid \omega_i \in Mod(K)\}$. We first show that for each $\omega \in Mod(K') \setminus \Omega'$, $A \subseteq \omega$. Indeed, for any triple $\omega_j, \omega_k, \omega_l \in Mod(K)$, such that $\omega_{jkl} = \mathrm{maj}_3(\omega_j, \omega_k, \omega_l) \notin Mod(K)$, we observe that $\mathrm{maj}_3(\omega'_j, \omega'_k, \omega'_l) = \omega_{jkl} \cup \mathrm{maj}_3(A \setminus \{a_j\}, A \setminus \{a_k\}, A \setminus \{a_l\}) = \omega_{jkl} \cup A$. Thus, for each $\omega \in Cl^1_{\mathcal{L}_{2CNF}}(\Omega') \setminus \Omega'$, $A \subseteq \omega$. Recall that $Mod(K') = Cl_{\mathcal{L}_{2CNF}}(\Omega')$. It follows quite easily that each further interpretation $\omega \in Cl_{\mathcal{L}_{2CNF}}(\Omega') \setminus (Cl^1_{\mathcal{L}_{2CNF}}(\Omega') \cup \Omega')$, also satisfies $A \subseteq \omega$.

This shows that each model of K' contains at least $n-1$ atoms from A. Thus, for every model $\omega_i \in K$, $H(\omega_i, K') = H(\omega_i, \omega'_i) = n-1$. It remains to show that for each $\omega \in Mod(\mu) \setminus Mod(K)$, $H(\omega, K') \geq n$. First, let $\omega' \in \Omega'$. Since $\omega \notin Mod(K)$, $\omega' \setminus A \neq \omega$ and since ω' contains $n-1$ elements from A, we have $H(\omega, \omega') \geq n$. As shown above all other interpretations $\omega'' \in Mod(K') \setminus \Omega'$ contain all n atoms from A, thus $H(\omega, \omega'') \geq n$, too. □

As an immediate consequence, we obtain that any KB K is $\mathcal{L}_{2CNF}$-distributable w.r.t. $\Delta^{H,\otimes}$ for *any* aggregation function $\otimes$. Note that this result is in strong contrast to the $\mathcal{L}_{1CNF}$ fragment, where only $\mathcal{L}_{1CNF}$-expressible KBs are $\mathcal{L}_{1CNF}$-distributable w.r.t. $\Delta^{H,\Sigma}$.

The Horn-Fragment

We now turn our attention to the $\mathcal{L}_{Horn}$ fragment. Recall Example 1 where we have shown how to distribute some non $\mathcal{L}_{Horn}$-expressible KB using a profile over two $\mathcal{L}_{Horn}$-KBs. Our first result shows that in this example case we cannot reduce to profiles of a single KB, i.e. that there are KBs which are $\mathcal{L}_{Horn}$-distributable but not $\mathcal{L}_{Horn}$-simplifiable.

Proposition 11. *Given a KB K with $Mod(K) = \{\omega_1, \omega_2, \omega_3\}$, where $\omega_3 = \omega_1 \cup \omega_2$, $H(\omega_1, \omega_2) = 2$ and ω_1, ω_2 are incomparable. Then K is not $\mathcal{L}_{Horn}$-simplifiable w.r.t. Δ^H.*

Proof. The situation described in the Proposition corresponds to $K = \{\omega \cup \{a\}, \omega \cup \{b\}, \omega \cup \{a,b\}\}$ with ω some interpretation which does not contain a or b. We need $Mod(\mu) = \{\omega, \omega \cup \{a\}, \omega \cup \{b\}, \omega \cup \{a,b\}\}$, as required by Proposition 2. We want to identify a $\mathcal{L}_{Horn}$-KB K' such that $\Delta^H_\mu(K') \equiv K$. This means that ω is the single model of μ which is not minimal w.r.t. the Hamming distance. Let ω'_1 be the model in K' closest to $\omega_1 = \omega \cup \{a\}$ and ω'_2 the one closest to $\omega_2 = \omega \cup \{b\}$. We need $a \in \omega'_1$ and $b \in \omega'_2$; otherwise $H(\omega, \omega'_1) < H(\omega_1, \omega'_1)$ or $H(\omega, \omega'_2) < H(\omega_2, \omega'_2)$; further we need $b \notin \omega'_1$ and $a \notin \omega'_2$; otherwise $H(\omega_3, \omega'_1) < H(\omega_1, \omega'_1)$ or $H(\omega_3, \omega'_2) < H(\omega_2, \omega'_2)$. Hence ω'_1 and ω'_2 are incomparable thus also $\omega'_1 \cap \omega'_2 \in Mod(K')$, since K' is a Horn KB. But then $H(\omega, \omega'_1 \cap \omega'_2) \leq H(\omega_1, \omega'_1)$. □

Our next result shows that Δ^H nonetheless increases the range of $\mathcal{L}_{Horn}$-simplifiable KBs compared to Δ^D (recall Theorem 5).

Proposition 12. *Any knowledge base K with $Mod(K) = \{\omega_1, \omega_2\}$ is $\mathcal{L}_{Horn}$-simplifiable w.r.t. Δ^H.*

Proof. If ω_1, ω_2 are comparable, we can apply Proposition 1. Thus, assume ω_1, ω_2 are incomparable and let $d_1 =$

$|\omega_1 \setminus \omega_2|$ and $d_2 = |\omega_2 \setminus \omega_1|$. W.l.o.g. assume $d_1 \leq d_2$. Also note that $d_1 > 0$. We use K' with $Mod(K') = \{\omega_1^+, \omega_1 \cup \omega_2\}$ where ω_1^+ adds d_1 elements from $\omega_2 \setminus \omega_1$ to ω_1. Thus, $\omega_1^+ \subseteq \omega_1 \cup \omega_2$ and we can choose K' from $\mathcal{L}_{Horn}$. Moreover, let $\mu \in \mathcal{L}_{Horn}$ such that $Mod(\mu) = \{\omega_1, \omega_2, \omega_1 \cap \omega_2\}$. We have the following distances (note that $d(\omega_2, \omega_1^+) = d_1 + (d_2 - d_1)$).

	ω_1^+	$\omega_1 \cup \omega_2$	K'
ω_1	d_1	d_2	d_1
ω_2	d_2	d_1	d_1
$\omega_1 \cap \omega_2$	$2d_1$	$d_1 + d_2$	$> d_1$

Hence, $\Delta_\mu^H(K') \equiv K$ as desired. □

Our final result concerns distributability in the *Horn* fragment. We show that some KBs with three models can be distributed.

Proposition 13. *Let K be a KB such that $Mod(K) = \{\omega_1, \omega_2, \omega_3\}$. If ω_1, ω_2 and ω_3 are not all pairwise incomparable, then K is Horn-distributable w.r.t. $\Delta^{H,\otimes}$ with $\otimes \in \{\Sigma, GMax, GMin\}$.*

Proof. If K is *Horn*-expressible, then the result follows from Proposition 1. If K is not *Horn*-expressible, then we do a case analysis on the number of pairwise incomparable models of K.

Case 1. If exactly one pair of models of K are incomparable, then we can assume without loss of generality that it is ω_1 and ω_2. It follows then that $\omega_3 \neq \omega_1 \cap \omega_2$. Also, there must be distinct atoms a and b such that $a \in \omega_1$, $a \notin \omega_2$ and $b \in \omega_2$, $b \notin \omega_1$. We consider a constraint $\mu \in \mathcal{L}_{Horn}$ such that $Mod(\mu) = \{\omega_1, \omega_2, \omega_3, \omega_1 \cap \omega_2\}$.

Case 1.1. If $\omega_1 \subseteq w_3$ and $w_2 \subseteq \omega_3$, then we take a globally new atom c and KBs K_1 and K_2 such that:

- $Mod(K_1) = \{\omega_1 \cup \{b\}, \omega_2 \cup \{a\}, \omega_3, (\omega_1 \cap \omega_2) \cup \{a, b\}\}$
- $Mod(K_2) = \{\omega_1, \omega_2, \omega_3 \cup \{c\}, \omega_1 \cap \omega_2\}$

It is easy to see that K_1 and K_2 are *Horn*-expressible. Considering, now, the profile $E = (K_1, K_2)$, we obtain the following distances:

	K_1	K_2	Σ	*GMax*	*GMin*
ω_1	1	0	1	$(1, 0)$	$(0, 1)$
ω_2	1	0	1	$(1, 0)$	$(0, 1)$
ω_3	0	1	1	$(1, 0)$	$(0, 1)$
$\omega_1 \cap \omega_2$	2	0	2	$(2, 0)$	$(0, 2)$

So for each $\oplus \in \{\Sigma, GMax, GMin\}$ we obtain that $\Delta_\mu^{H,\oplus}(E) \equiv K$.

Case 1.2. If $\omega_3 \subseteq \omega_1$ and $\omega_3 \subseteq \omega_2$, then $\omega_3 \subseteq \omega_1 \cap \omega_2$. Moreover, since $\omega_3 \neq \omega_1 \cap \omega_2$, it actually holds that $\omega_3 \subset \omega_1 \cap \omega_2$. Thus there exists an atom c such that $c \in (\omega_1 \cap \omega_2)$ and $c \notin \omega_3$. We now take KBs K_1, K_2, K_3 and K_4 such that:

- $Mod(K_1) = \{\omega_1 \cup \{b\}, \omega_2 \cup \{a\}, \omega_3, (\omega_1 \cap \omega_2) \setminus \{c\}, (\omega_1 \cap \omega_2) \cup \{a, b\}\}$
- $Mod(K_2) = \{\omega_1, \omega_2 \cup \{a\}, \omega_3, (\omega_1 \cap \omega_2) \cup \{a\}\}$
- $Mod(K_3) = \{\omega_1 \cup \{b\}, \omega_2, \omega_3, (\omega_1 \cap \omega_2) \cup \{b\}\}$
- $Mod(K_4) = \{\omega_1, \omega_2, \omega_3 \cup \{c\}, \omega_1 \cap \omega_2\}$

It is easy to see that K_1, K_2, K_3 and K_4 are *Horn*-expressible. Considering, now, the profile $E = (K_1, K_2, K_3, K_4, K_4)$, we obtain the following distances:

	K_1	K_2	K_3	K_4	K_4	Σ	*GMax*	*GMin*
ω_1	1	0	1	0	0	2	(1,1,0,0,0)	(0,0,0,1,1)
ω_2	1	1	0	0	0	2	(1,1,0,0,0)	(0,0,0,1,1)
ω_3	0	0	0	1	1	2	(1,1,0,0,0)	(0,0,0,1,1)
$\omega_1 \cap \omega_2$	1	1	1	0	0	3	(1,1,1,0,0)	(0,0,1,1,1)

So for each $\oplus \in \{\Sigma, GMax, GMin\}$ we obtain that $\Delta_\mu^{H,\oplus}(E) \equiv K$.

Case 2. If exactly two pairs of models of K are incomparable, then we can assume without loss of generality that it is w_1, w_2 and w_2, w_3. We consider a constraint $\mu \in \mathcal{L}_{Horn}$ such that $Mod(\mu) = \{\omega_1, \omega_2, \omega_3, \omega_1 \cap \omega_2, \omega_2 \cap \omega_3\}$. Then there must be distinct atoms a and b such that $a \in \omega_1$, $a \notin \omega_2$ and $b \in \omega_2$, $b \notin \omega_1$. Further, there must be distinct atoms c and d such that $c \in \omega_2$, $c \notin \omega_3$ and $d \in \omega_3$, $d \notin \omega_2$.

Case 2.1. If $w_1 \subseteq w_3$, then we get that $c \notin \omega_1$ and $a \in \omega_3$. We take KBs K_1 and K_2 such that:

- $Mod(K_1) = \{\omega_1 \cup \{c\}, \omega_2, \omega_3 \cup \{c\}, (\omega_1 \cap \omega_2) \cup \{c\}, (\omega_2 \cap \omega_3) \cup \{c\}\}$
- $Mod(K_2) = \{\omega_1, \omega_2 \cup \{a\}, \omega_3, (\omega_1 \cap \omega_2) \cup \{a\}, (\omega_2 \cap \omega_3) \cup \{a\}\}$

It is easy to see that K_1 and K_2 are *Horn*-expressible. Considering, now, the profile $E = (K_1, K_2)$ and keeping in mind that $c \notin \omega_1$ and $a \in \omega_3$, we obtain the following distances:

	K_1	K_2	Σ	*GMax*	*GMin*
ω_1	1	0	1	$(1, 0)$	$(0, 1)$
ω_2	0	1	1	$(1, 0)$	$(0, 1)$
ω_3	1	0	1	$(1, 0)$	$(0, 1)$
$\omega_1 \cap \omega_2$	1	1	2	$(1, 1)$	$(1, 1)$
$\omega_2 \cap \omega_3$	1	1	2	$(1, 1)$	$(1, 1)$

So for each $\oplus \in \{\Sigma, GMax, GMin\}$ we obtain that $\Delta_\mu^{H,\oplus}(E) \equiv K$.

Case 2.2. If $w_3 \subseteq w_1$, then we get that $b \notin \omega_3$ and $d \in \omega_1$. We take KBs K_1, K_2 and such that:

- $Mod(K_1) = \{\omega_1 \cup \{b\}, \omega_2, \omega_3 \cup \{b\}, (\omega_1 \cap \omega_2) \cup \{b\}, (\omega_2 \cap \omega_3) \cup \{b\}\}$
- $Mod(K_2) = \{\omega_1, \omega_2 \cup \{d\}, \omega_3, (\omega_1 \cap \omega_2) \cup \{d\}, (\omega_2 \cap \omega_3) \cup \{d\}\}$

It is easy to see that K_1 and K_2 are *Horn*-expressible. Considering, now, the profile $E = (K_1, K_2)$ and keeping in mind that $b \notin \omega_3$ and $d \in \omega_1$, we obtain the following distances:

	K_1	K_2	Σ	*GMax*	*GMin*
ω_1	1	0	1	$(1, 0)$	$(0, 1)$
ω_2	0	1	1	$(1, 0)$	$(0, 1)$
ω_3	1	0	1	$(1, 0)$	$(0, 1)$
$\omega_1 \cap \omega_2$	1	1	2	$(1, 1)$	$(1, 1)$
$\omega_2 \cap \omega_3$	1	1	2	$(1, 1)$	$(1, 1)$

	1CNF	*2CNF*	*Horn*
simplifiable w.r.t. Δ^{D}	×	×	×
simplifiable w.r.t. Δ^{H}	×	✓	○
distributable w.r.t. $\Delta^{D,\Sigma}$	✓	✓	✓
distributable w.r.t. $\Delta^{H,\Sigma}$	×	✓	−
distributable w.r.t. $\Delta^{H,GMax}$	−	✓	−
distributable w.r.t. $\Delta^{H,GMin}$	−	✓	−

Table 1: Summary of Results

So for each $\oplus \in \{\Sigma, GMax, GMin\}$ we obtain that $\Delta_{\mu}^{H,\oplus}(E) \equiv K$. The cases when $\omega_2 \subseteq \omega_3$ or $\omega_3 \subseteq \omega_2$ are symmetric. This concludes our case analysis, as any other remaining case results in either all of the interpretations ω_1, ω_2 and ω_3 being pairwise incomparable, or in K being *Horn*-expressible. □

The remaining case (i.e., $Mod(K) = \{\omega_1, \omega_2, \omega_3\}$ with $\omega_1, \omega_2, \omega_3$ pairwise incomparable), as well as the more general case when K has an arbitrary number of models is subject to ongoing work.

Conclusion

In this paper we have proposed the notion of distributability and we have studied the properties of several merging operators with respect to different fragments of propositional logic. Our results are summarized in Table 1. The symbol × means that only "trivial" knowledge bases (belonging to the considered fragment) can be distributed with the corresponding operator. Alternately, ✓ means that any knowledge base can be distributed. Symbol − means we know that some non-trivial knowledge bases can be distributed, and finally ○ means that some, but not all, non-trivial bases can be simplified. Interestingly, the picture emerging from Table 1 is that merging operators behave quite differently depending on the distance and aggregation function employed, in a way that does not lend itself to simple categorization. For instance, our results on simplifiability imply that using Dalal revision to $\mathcal{L}_{\mathit{1CNF}}$ KBs never takes us outside the *1CNF* fragment; applying the same revision operator to $\mathcal{L}_{\mathit{2CNF}}$ KBs can produce any KB in $\mathcal{L}$; and applying it to $\mathcal{L}_{Horn}$ KBs can produce some, though not all possible KBs.

Several questions are still open for future work. We plan to study the exact characterization of what can (and cannot) be distributed, in order to replace the symbols − and ○ in the previous table. Other merging operators can also be integrated to our study. Some of our results on distributability require the addition of new atoms to the interpretations. We want to determine whether similar results can be obtained without modifying the set of propositional variables, in particular for the *2CNF* fragment. We are also interested in the number of knowledge bases needed to distribute knowledge: given an integer n, a knowledge base K and a merging operator Δ, is it possible to distribute K w.r.t. Δ such that the resulting profile contains at most n knowledge bases? This paper was a first step to understand the limits of distributability; the actual construction of the profile and complexity of this process are important questions that will be tackled in future research. Finally, we also consider applying the concept of distributability to non-classical formalisms, in particular in connection with merging operators proposed for logic programs (Delgrande et al. 2013).

Acknowledgments

This work was supported by the Austrian Science Fund (FWF) under grant P25521.

References

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change : Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50:510–530.

Creignou, N.; Papini, O.; Pichler, R.; and Woltran, S. 2014a. Belief revision within fragments of propositional logic. *Journal of Computer and System Sciences* 80(2):427–449.

Creignou, N.; Papini, O.; Rümmele, S.; and Woltran, S. 2014b. Belief merging within fragments of propositional logic. In *Proceedings of the Twenty-First European Conference on Artificial Intelligence (ECAI'14)*, 231–236.

Creignou, N.; Pichler, R.; and Woltran, S. 2013. Do hard SAT-related reasoning tasks become easier in the Krom fragment? In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI'13)*, 824–831.

Darwiche, A., and Marquis, P. 2002. A knowledge compilation map. *Journal of Artificial Intelligence Research (JAIR)* 17:229–264.

Delgrande, J. P., and Peppas, P. 2015. Belief revision in horn theories. *Artificial Intelligence* 218:1–22.

Delgrande, J. P.; Schaub, T.; Tompits, H.; and Woltran, S. 2013. A model-theoretic approach to belief change in answer set programming. *ACM Transactions on Computational Logic* 14(2).

Eiter, T., and Gottlob, G. 1992. On the complexity of propositional knowledge base revision, updates, and counterfactuals. *Artificial Intelligence* 57(2-3):227–270.

Fargier, H., and Marquis, P. 2014. Disjunctive closures for knowledge compilation. *Artificial Intelligence* 216:129–162.

Haret, A.; Rümmele, S.; and Woltran, S. 2015. Merging in the Horn fragment. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI'15)*, 3041–3047.

Katsuno, H., and Mendelzon, A. O. 1991. Propositional knowledge base revision and minimal change. *Artificial Intelligence* 52:263–294.

Konieczny, S., and Pino Pérez, R. 2002. Merging information under constraints: a logical framework. *Journal of Logic and Computation* 12(5):773–808.

Konieczny, S.; Lang, J.; and Marquis, P. 2002. Distance-based merging: a general framework and some complexity results. In *Proceedings of the Eighth International Conference on Principles of Knowledge Representation and Reasoning (KR'02)*, 97–108.

Konieczny, S.; Lang, J.; and Marquis, P. 2004. DA^2 merging operators. *Artificial Intelligence* 157(1-2):49–79.

Liberatore, P., and Schaerf, M. 2001. Belief revision and update: Complexity of model checking. *Journal of Computer and System Sciences* 62(1):43–72.

Liberatore, P. 2015a. Belief merging by examples. *ACM Transactions on Computational Logic* 17(2):9:1–9:38.

Liberatore, P. 2015b. Revision by history. *Journal of Artificial Intelligence Research (JAIR)* 52:287–329.

Marquis, P. 2015. Compile! In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 4112–4118.

Zhuang, Z. Q., and Pagnucco, M. 2012. Model based horn contraction. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning (KR'12)*, 169–178.

Zhuang, Z., and Pagnucco, M. 2014. Entrenchment-based horn contraction. *Journal of Artificial Intelligence Research (JAIR)* 51:227–254.

Zhuang, Z. Q.; Pagnucco, M.; and Zhang, Y. 2013. Definability of horn revision from horn contraction. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI'13)*, 1205–1211.

Relations between assumption-based approaches in nonmonotonic logic and formal argumentation[*]

Jesse Heyninck and **Christian Straßer**
Institute of Philosophy II, Ruhr Universität Bochum
Universitätstraße 150
44 800 Bochum, Germany

Abstract

In this paper we make a contribution to the unification of formal models of defeasible reasoning. We present several translations between formal argumentation frameworks and nonmonotonic logics for reasoning with plausible assumptions. More specifically, we translate adaptive logics into assumption-based argumentation and ASPIC$^+$, ASPIC$^+$ into assumption-based argumentation and a fragment of assumption-based argumentation into adaptive logics. Adaptive logics are closely related to Makinson's default assumptions and to a significant class of systems within the tradition of preferential semantics in the vein of KLM and Shoham. Thus, our results also provide close links between formal argumentation and the latter approaches.

1 Introduction

There is a a plenitude of logical approaches to the modelling of defeasible reasoning known as nonmonotonic logics (in short, NMLs). These approaches often use different methods, representational formats or key ideas, making it sometimes difficult to compare them, e.g. with respect to the consequence relations they give rise to. Such comparisons are important to systematise the field of NMLs and to gain insights into which forms of defeasible reasoning are expressible in which formal frameworks. An important tool for such comparisons are translations between systems of NML. If one system (or a fragment thereof) is translatable into another system we immediately know that the latter system is at least as expressive as the former. Moreover, this may lead to forms of cross-fertilisation, since meta-theoretic properties become transferable between the translated systems.

In this contribution we will investigate several such translations. Given the richness of the domain of NMLs, we approach the topic from a specific angle. Our focus will be on structured argumentation, on the one hand, and NMLs that model defeasible inferences in terms of strict inference rules and defeasible assumptions, on the other hand. As a side product, the translation will also cover a significant subclass of NMLs in the KLM paradigm based on preferential semantics [18, 12].

At least since Dung introduced abstract argumentation [8], formal argumentation has been an important sub-domain of NML. While in abstract argumentation arguments are not phrased in a formal language and the underlying inferences are not explicated, several systems of structured or instantiated formal argumentation have been developed which overcome this limitation (cf. [5] for a partial overview). In this paper we will focus on two of the most prominent accounts: assumption-based argumentation (in short, ABA) [7, 10, 20] and ASPIC$^+$ [16, 14].

One of the key differences between several formal approaches to defeasible reasoning concerns the question of how to model defeasible inferences. Let $A_1, \ldots, A_n \rightsquigarrow B$ denote the defeasible inference from $A_1, \ldots, A_n$ to B. The question is whether such an inference should be phrased in terms of a strict inference rule or a defeasible one. A strict inference rule allows for no exceptions: if its premises $A_1, \ldots, A_n$ are true, the consequent B is true as well. In contrast, defeasible rules allow for exceptions, that is, under specific circumstances it may hold that all premises $A_1, \ldots, A_n$ of the rule hold while the consequent B doesn't. Clearly, in the approach with strict rules defeasibility has to enter in a different way. One way is by means of explicitly stated defeasible assumptions $\mathrm{As}_1, \ldots, \mathrm{As}_m$, i.e., specific premises which are assumed to hold by default and which can serve as antecedents of strict rules. An inference is retracted in case there is a demonstration that one of the defeasible assumptions $\mathrm{As}_1, \ldots, \mathrm{As}_m$ doesn't hold.

ABA follows the approach based on strict rules and defeasible assumptions. In ASPIC$^+$ both approaches can be represented. Not surprisingly, ABA has been shown to be translatable to ASPIC$^+$ [16]. In this paper we will show the other (perhaps more surprising) direction, namely that ASPIC$^+$ (without priorities) can be translated into ABA and thus that both frameworks are equi-expressive.

There are several nonmonotonic systems that model defeasible inference by means of strict rules. Among them are adaptive logics (in short, ALs) [4], Makinsons' default assumptions and forms of circumscription. Makinson's default assumptions –and in view of the translation in [22] also ALs– are a generalisations of approaches based on maximal

[*]The research of the authors was supported by a Sofja Kovalevkaja award of the Alexander von Humboldt-Foundation, funded by the German Ministry for Education and Research.

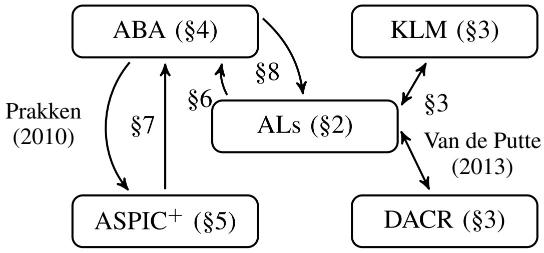


Figure 1: Roadmap

consistent subsets [17]. In view of [1] we know that there are close connections between approaches based on maximal consistent subsets and structured argumentation. In this paper the ties will be strengthened. We show that ALs can be translated into ABA and ASPIC$^+$ and present a translation in the opposite direction for a subclass of ABA and ASPIC$^+$.

We will proceed as follows: in Sections 2–5 we introduce the basic systems (ALs, preferential semantics, default assumptions, ABA, and ASPIC$^+$). In Sections 6–8 we provide translations as indicated in Figure 1.

2 Adaptive Logics

ALs are a general framework for the formal explication of defeasible reasoning. It has been applied to a multitude of defeasible reasoning forms (mainly related to questions from philosophical logic), such as nonmonotonic forms of reasoning with inconsistent information, causal discovery, inductive generalisations, abductive hypothesis generation, normative reasoning, etc. (see [19, p.86] for an overview).

The driving idea behind ALs is to apply defeasible inference rules under explicit normality assumptions. More specifically, given a compact Tarksi logic $\mathbf{L}$ (the *core* or *lower limit logic*) in a formal language $\mathcal{L}$ and with the derivability relation $\vdash_{\mathbf{L}}$, a set of *abnormalities* $\Omega \subseteq \mathcal{L}$ is fixed. Now, whenever the core logic gives rise to $\Gamma \vdash_{\mathbf{L}} A \vee \mathsf{ab}$ where $\mathsf{ab} \in \Omega$, A can be derived in the adaptive logic (based on $\mathbf{L}$ and Ω) on the (defeasible) assumption that ab is false.[1]

In ALs this basic idea of modeling defeasible inferences is implemented in Hilbert-style proofs. We will first explain the proof theory of ALs.[2] Then we give alternative characterizations of the adaptive consequence relations that are central to prove the adequacy of our translations in subsequent sections.

In ALs, usual Hilbert-style proofs are adjusted in two major ways. First, to keep track of normality assumptions, proof lines in adaptive proofs are equipped with an additional column in which the abnormalities are listed that are assumed to be false. Second, different *retraction mechanisms* for lines with abnormality assumptions that turn out mistaken are implemented in terms of so-called *adaptive strategies*. We will give some examples below.

To further explain how adaptive proofs work, it is useful to turn to a concrete example. As an illustration, we take a look at inconsistency-ALs. These are based on paraconsistent core logics such as **LP** or **CLuN(s)**[3]. These core logics typically do not validate disjunctive syllogism $A, {\sim}A \vee B \vdash B$ since in case A is involved in a contradiction, B would not follow (then ${\sim}A$ would suffice for the disjunction ${\sim}A \vee B$ to be true). Nevertheless, inconsistency-ALs allow for the defeasible application of disjunctive syllogism under the *normality assumption* that there is no contradiction in A. Hence, in inconsistency ALs the abnormalities in Ω typically have the form of a contradiction $A \wedge {\sim}A$. E.g., in paraconsistent core logics it usually holds that $A, {\sim}A \vee B \vdash B \vee (A \wedge {\sim}A)$ and thus one can defeasibly derive B under the assumption that there is no contradiction in A. Clearly, sometimes such assumptions turn out to be mistaken in view of the given premises. Obviously, this is the case if $A \wedge {\sim}A$ is derivable from the given premises. A more interesting case is given, if $A \wedge {\sim}A$ is not directly derivable but it is derivable as a member of a minimal disjunction of abnormalities. We illustrate this in the following example.

Example 1. *Suppose our core logic is a standard paraconsistent logic such as* $\mathbf{LP}$ *or* $\mathbf{CLuN(s)}$. *Let* $\Gamma = \{{\sim}p, {\sim}q, p \vee q, p \vee r, q \vee s\}$.

1	${\sim}p$	PREM	$\emptyset$
2	${\sim}q$	PREM	$\emptyset$
3	$p \vee r$	PREM	$\emptyset$
4	$q \vee s$	PREM	$\emptyset$
5	$p \vee q$	PREM	$\emptyset$
6	$r \vee (p \wedge {\sim}p)$	1,3,**L**-Inf	$\emptyset$
7	$s \vee (q \wedge {\sim}q)$	2,4,**L**-Inf	$\emptyset$
8	r	6,RC	$\{p \wedge {\sim}p\}$
9	s	7,RC	$\{q \wedge {\sim}q\}$
10	$r \vee s$	8,**L**-Inf	$\{p \wedge {\sim}p\}$
11	$r \vee s$	9,**L**-Inf	$\{q \wedge {\sim}q\}$
12	$(p \wedge {\sim}p) \vee (q \wedge {\sim}q)$	1,2,5,**L**-Inf	$\emptyset$

Each proof line has 4 elements: a line number, a formula, a justification and a set of abnormalities (which are assumed to be false). All inferences of the core logic **L** can be applied (indicated by **L**-Inf in lines 6, 10, 11 and 12). In lines 8 and 9 defeasible inferences are made as explained above. E.g., since at line 6 $r \vee (p \wedge {\sim}p)$ is derived, at line 8 the abnormality $p \wedge {\sim}p$ is considered false and thus put in the abnormality column. The rule employed for this is called RC (rule conditional): from $(l; A \vee \mathsf{ab}; \Delta)$ derive $(l'; A; l, RC; \Delta \cup \{\mathsf{ab}\})$. When further inferences are made calling upon lines with non-empty sets of abnormalities, these abnormalities are carried over (see lines 10 and 11 where the abnormalities of lines 8 and 9 are carried over).

The retraction of lines in adaptive proofs is always determined in view of the minimal disjunctions of abnormalities derived at a given stage of a proof (on the empty set of abnormalities). At line 12 such a minimal disjunction of ab-

[1] The disjunction $\vee$ is supposed to be classical. In fact, in the standard format of ALs which we consider here, the core logic is supposed to be supraclassical. Whenever non-classical logics are used as core logics, classical negation $\neg$ and classical disjunction $\vee$ are superimposed.

[2] Due to spatial restrictions we will focus on the main ideas but explain some aspects of the proof theory (such as adaptive strategies) merely in a semi-formal way. For a more thorough explanation the interested reader is referred to [4, 19].

[3] **CLuN(s)** is positive classical logic enriched by the law of the excluded middle. For an axiomatization and a semantics see [3].

normalities is derived. Clearly, the abnormalities assumed to be false at lines 8–11 are involved in the given disjunction. There are different retraction mechanisms for ALs: so-called adaptive *strategies*. According to the *reliability* strategy, any line with an abnormality in the assumption that is part of a minimal disjunction of abnormalities is to be retracted. Retraction is implemented by marking lines that are retracted. In this case:

✓	8	r	6,RC	$\{p \wedge {\sim}p\}$
✓	9	s	7,RC	$\{q \wedge {\sim}q\}$
✓	10	$r \vee s$	8,**L**-Inf	$\{p \wedge {\sim}p\}$
✓	11	$r \vee s$	9,**L**-Inf	$\{q \wedge {\sim}q\}$

There are other, less cautious, strategies. For instance, according to the *minimal abnormality strategy*, $r \vee s$ will not be retracted. The reason is as follows. If we interpret our premises strictly *as normal as possible*, then in view of line 12 it will be the case that either $p \wedge {\sim}p$ holds (and $q \wedge {\sim}q$ doesn't), or $q \wedge {\sim}q$ holds (and $p \wedge {\sim}p$ doesn't). In each case, one of the assumptions of line 10 or 11 is warranted. Due to space limitations, we omit the technical details. Yet another strategy is *normal selections*. According to it a line with the set of abnormalities Δ is retracted (or marked) once $\bigvee \Delta$ is derived on the empty condition.

These retraction mechanisms provided by adaptive strategies make AL proofs dynamic: sometimes a line may get marked, later unmarked, and yet later marked again. In order to define a *consequence relation* we need a stable notion of derivability. It works as follows: a formula at a line l of a proof is *finally derived* at a stage of the proof if l is not marked and every extension of the proof in which it gets marked can be further extended such that it is unmarked again. The consequence relation of ALs is the defined as follows:

Definition 1. *Let* $\mathbf{L}$ *be a compact Tarski logic in the formal language* $\mathcal{L}$*, let* $\Omega \subseteq \mathcal{L}$ *be a set of abnormalities, and let* $\mathsf{str} \in \{\mathsf{r}, \mathsf{ma}, \mathsf{ns}\}$ *be an adaptive strategy (reliability, minimal abnormality, or normal selections). Where* $\Gamma \cup \{A\} \subseteq \mathcal{L}$*,* $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{str}} A$ *iff* A *is finally derivable in an adaptive proof from* Γ*.*

For our translations alternative characterisations of the consequence relations defined in terms of final derivability in Definition 1 will be very useful. These characterisations are essentially informed by the set of minimal disjunctions of abnormalities derivable from a given premise set by the core logic $\mathbf{L}$.

Definition 2. *Where* $\Gamma \subseteq \mathcal{L}$*:* $\Sigma_{\mathbf{L}}(\Gamma)$ *is the set of all non-empty* $\Delta \subseteq \Omega$ *such that* $\Gamma \vdash_{\mathbf{L}} \bigvee \Delta$ *and for all non-empty* $\Delta' \subset \Delta$*,* $\Gamma \nvdash_{\mathbf{L}} \bigvee \Delta'$*.*

A choice set over $\Sigma_{\mathbf{L}}(\Gamma)$ is a set Θ for which $\Delta \cap \Theta \neq \emptyset$ for all $\Delta \in \Sigma_{\mathbf{L}}(\Gamma)$.

Definition 3. *Where* $\Gamma \subseteq \mathcal{L}$*:* $\Phi_{\mathbf{L}}(\Gamma)$ *is the set of* $\subset$*-minimal choice sets over* $\Sigma_{\mathbf{L}}(\Gamma)$*.*

The following facts will be useful in what follows:

Fact 1 ([19]). *1. For all choice sets* Θ *over* $\Sigma_{\mathbf{L}}(\Gamma)$ *there is a* $\Theta' \in \Phi_{\mathbf{L}}(\Gamma)$ *such that* $\Theta' \subseteq \Theta$*.*

2. $\phi \in \Phi_{\mathbf{L}}(\Gamma)$ *iff* ϕ *is a choice set of* $\Sigma_{\mathbf{L}}(\Gamma)$ *and for all* $A \in \phi$ *there is a* $\Delta_A \in \Sigma_{\mathbf{L}}(\Gamma)$ *for which* $\{A\} = \Delta_A \cap \phi$*.*

We now give representation theorems for all three adaptive strategies, a given core logic $\mathbf{L}$ and a given set of abnormalities Ω.

Theorem 1 ([4]). $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{ma}} A$ *iff for all* $\Theta \in \Phi_{\mathbf{L}}(\Gamma)$ *there is a* $\Delta \subseteq \Omega \setminus \Theta$ *such that* $\Gamma \vdash_{\mathbf{L}} A \vee \bigvee \Delta$*.*

Theorem 2 ([4]). $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{r}} A$ *iff there is a* $\Delta \subseteq \Omega \setminus \bigcup \Sigma_{\mathbf{L}}(\Gamma)$ *such that* $\Gamma \vdash_{\mathbf{L}} A \vee \bigvee \Delta$*.*

Theorem 3 ([4]). $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{ns}} A$ *iff there is a* $\Theta \in \Phi_{\mathbf{L}}(\Gamma)$ *and a* $\Delta \subseteq \Omega \setminus \Theta$ *such that* $\Gamma \vdash_{\mathbf{L}} A \vee \bigvee \Delta$*.*

3 Preferential Semantics and Default Assumptions

The semantics for ALs are a special but rich subclass of the well known preferential semantics as defined in [12] and [18]. As in the previous section we assume a core logic $\mathbf{L}$ in a formal language $\mathcal{L}$ and a set of abnormalities $\Omega \subseteq \mathcal{L}$. We also assume that the core logic $\mathbf{L}$ comes with an adequate model-theoretic semantics and an associated semantic consequence relation $\Vdash_{\mathbf{L}}$. We write $\mathcal{M}(\Gamma)$ for the set of all models of a premise set Γ. Furthermore, where $M \in \mathcal{M}(\Gamma)$, $Ab(M) = \{A \in \Omega \mid M \models A\}$. A model $M \in \mathcal{M}(\Gamma)$ is *minimally abnormal* iff there is no $M' \in \mathcal{M}(\Gamma)$ for which $Ab(M') \subset Ab(M)$.

Definition 4.
- $\Gamma \Vdash^{\Omega,\mathbf{L}}_{\mathsf{ma}} A$ *iff* $M \models A$ *for every minimally abnormal model of* Γ*.*
- $\Gamma \Vdash^{\Omega,\mathbf{L}}_{\mathsf{r}} A$ *iff* $M \models A$ *for every* $M \in \mathcal{M}(\Gamma)$ *for which all* $A \in Ab(M)$ *are verified in some minimally abnormal model* $M' \in \mathcal{M}(\Gamma)$*.*
- $\Gamma \Vdash^{\Omega,\mathbf{L}}_{\mathsf{ns}} A$ *iff there is a minimally abnormal model* $M \in \mathcal{M}(\Gamma)$ *such that for all* $M' \in \mathcal{M}(\Gamma)$ *for which* $Ab(M) = Ab(M')$*,* $M' \models A$*.*

ALs in the standard format are sound and complete w.r.t. these semantics (proven e.g. in [4]):

Theorem 4. *Where* $\Gamma \cup \{A\} \subseteq \mathcal{L}$ *and* $\mathsf{str} \in \{\mathsf{ma}, \mathsf{r}, \mathsf{ns}\}$*,* $\Gamma \Vdash^{\Omega,\mathbf{L}}_{\mathsf{str}} A$ *iff* $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{str}} A$*.*

In [22], the connection between ALs and Makinson's Default Assumption Consequence Relations (in short, DACRs) [13, chapter 2] was established. In [13, chapter 2], it is also shown that many other non-monotonic consequence relations, such as Reiter's Closed World Assumption, Poole's Background Constraints, etc. can be expressed as DACRs. DACRs give formal substance to the idea that, in many situations, non-monotonic reasoning makes use of a set Δ of defeasible background assumptions in combination with the strict and explicit premises in Γ. These background assumptions are used to the extent that they are consistent with Γ. Accordingly, DACRs make use of the notion of maximal consistent subset:

Definition 5. *Where* $\Gamma \cup \Delta \subseteq \mathcal{L}$*,* $\Theta \subseteq \Delta$ *is a maximal* Γ*-consistent subset of* Δ *iff:*

- $\Gamma \cup \Theta \nvdash_{\mathbf{L}} A$ *for some* $A \in \mathcal{L}$ *and*
- $\Gamma \cup \Theta' \vdash_{\mathbf{L}} A$ *for all* $A \in \mathcal{L}$ *and for every* $\Theta \subset \Theta' \subseteq \Delta$*.*

$\mathsf{MCS}(\Gamma, \Delta)$ *is the set of all maximal* Γ*-consistent subsets of* Δ.

Definition 6. *Where* $\Gamma \cup \Delta \cup \{A\} \subseteq \mathcal{L}$, $\Gamma \vdash^{\mathrm{DA},\mathbf{L}}_{\Delta} A$ *iff for every* $\Delta' \in \mathsf{MCS}(\Gamma, \Delta)$, $\Gamma \cup \Delta' \vdash_{\mathbf{L}} A$.

The connection between adaptive logic and DACR's is the following:

Theorem 5. *[22, p.10] Where* $\Gamma \cup \Delta \cup \{A\} \subseteq \mathcal{L}$ *and* $\Delta^{\neg} = \{\neg B \mid B \in \Delta\}$, $\Gamma \vdash^{\mathrm{DA},\mathbf{L}}_{\Delta} A$ *iff* $\Gamma \vdash^{\Delta^{\neg},\mathbf{L}}_{\mathsf{ma}} A$.

4 Assumption-Based Argumentation

ABA, thoroughly described in [7], is a formal model that allows one to use a set of plausible assumptions "to extend a given theory" [7, p.70] unless and until there are good arguments for not using such an assumption.

Inferences are implemented in ABA by means of a deductive system consisting of a language and rules formulated over this language:

Definition 7 (Deductive System)**.** *A* deductive system *is a pair* $(\mathcal{L}, \mathcal{R})$ *such that*

- $\mathcal{L}$ *is a formal language (consisting of countably many sentences).*
- $\mathcal{R}$ *is a set of inference rules of the form* $A_1, \ldots, A_n \to A$ *and* $\to A$*, where* $A, A_1 \ldots, A_n \in \mathcal{L}$

Definition 8. *An* $\mathcal{R}$-deduction *from a theory* Γ *is a sequence* $B_1, \ldots, B_m$*, where* $m > 0$ *such that for all* $i = 1, \ldots, m$*:* $B_i \in \Gamma$ *or there exists a* $A_1, \ldots, A_n \to B_i \in \mathcal{R}$ *such that* $A_1, \ldots, A_n \in \{B_1, \ldots, B_{i-1}\}$.

Definition 9. *Where* $\Gamma \cup \{A\} \subseteq \mathcal{L}$, $\Gamma \vdash_{\mathcal{R}} A$ *holds if there is an* $\mathcal{R}$*-deduction from* Γ *whose last element is* A.

We now introduce defeasible assumptions and a contrariness operator to express argumentative attacks. Given a rule system, an assumption-based framework is defined as follows:

Definition 10 (Assumption-based framework)**.** *An* assumption-based framework *is a tuple* $\mathbf{ABF} = ((\mathcal{L}, \mathcal{R}), \Gamma, Ab, \overline{})$ *where:*

- $(\mathcal{L}, \mathcal{R})$ *is a deductive system*
- $\Gamma \subseteq \mathcal{L}$
- $\emptyset \neq Ab \subseteq \mathcal{L}$ *is the set of candidate assumptions.*
- $\overline{} : Ab \to \mathcal{L}$ *is a contrariness operator.*[4]

In most structured accounts of argumentation attacks are defined between arguments which are deductions in a given deductive or defeasible system (e.g., in ASPIC$^+$, Defeasible Logic Programming [11]) or sequents $\Gamma \vdash_{\mathbf{L}} A$ where $\mathbf{L}$ is an underlying core logic ([2, 6]).[5] In contrast, ABA operates at a higher level of abstraction, since attacks are defined directly on the level of sets of assumptions instead of on the level of $\mathcal{R}$-deductions.[6] ABA can thus be viewed as operating on the level of equivalence classes consisting of arguments generated using the same assumptions.

Definition 11 (Attacks)**.** *Given an assumption-based framework* $\mathbf{ABF} = ((\mathcal{L}, \mathcal{R}), \Gamma, Ab, \overline{})$*:*

- *a set of assumptions* $\Delta \subseteq Ab$ attacks *an assumption* $A \in Ab$ *iff* $\Gamma \cup \Delta \vdash_{\mathcal{R}} \overline{A}$.
- *a set of assumptions* $\Delta \subseteq Ab$ attacks *a set of assumptions* $\Delta' \subseteq Ab$ *iff* $\Gamma \cup \Delta \vdash_{\mathcal{R}} \overline{A}$ *for some* $A \in \Delta'$.

Consequences of a given assumption-based framework are determined with the use of argumentation semantics. On the basis of argumentative attacks, semantics determine sets of assumptions that are acceptable given different criteria of acceptability, such as the requirement that a given set of assumption should not attack itself, or it should be able to defend itself against attacks by other sets of assumptions. Argumentation semantics have been phrased for abstract frameworks in [8] and have been generalised to the level of ABA in e.g. [7].

Definition 12 (Argumentation semantics)**.** *Where* $\Delta \subseteq Ab$*:*

- Δ *is* closed *iff* $\Delta = \{A \in Ab \mid \Gamma \cup \Delta \vdash_{\mathcal{R}} A\}$.
- Δ *is* conflict-free *iff for every* $A \in Ab, \Delta \cup \Gamma \not\vdash_{\mathcal{R}} A$ *or* $\Delta \cup \Gamma \not\vdash_{\mathcal{R}} \overline{A}$.
- *A closed set* Δ *is* naive *iff it is maximally (w.r.t. set inclusion) conflict-free.*
- *A closed set of assumptions* $\Delta \subseteq Ab$ *is* admissible *iff it is conflict-free and for each closed set of assumptions* $\Delta' \subseteq Ab$*, if* Δ' *attacks* Δ*, then* Δ *attacks* Δ'.
- *A set* Δ *is* preferred *iff it is maximally (w.r.t. set inclusion) admissible.*
- Δ *is* stable *iff it is closed, conflict-free and attacks every* $a \in Ab \setminus \Delta$.

We write $\mathsf{niv}(\mathbf{ABF})$, $\mathsf{prf}(\mathbf{ABF})$ *resp.* $\mathsf{stb}(\mathbf{ABF})$ *for the set of naive, preferred resp. stable sets of assumptions in* $\mathbf{ABF}$.

Example 2. *Let* $Ab = \{q, \neg p \vee \neg q\}$, $\Gamma = \{p\}$*, let the rule system* $\mathcal{R}$ *characterize classical logic and* $\overline{A} = \neg A$ *(where* $\neg$ *is classical negation). Then there are two preferred sets:* $\{\neg p \vee \neg q\}, \{q\}$*. To see this note that e.g.* $\Gamma \cup \{\neg p \vee \neg q\} \vdash_{\mathcal{R}} \neg q$ *and* $\Gamma \cup \{q\} \vdash_{\mathcal{R}} \neg(\neg p \vee \neg q)$.

We are now in a position to define various consequence relations for ABA:

Definition 13. *Given an assumption-based framework* $\mathbf{ABF} = ((\mathcal{L}, \mathcal{R}), \Gamma, Ab, \overline{})$ *and* $\mathsf{sem} \in \{\mathsf{niv}, \mathsf{prf}, \mathsf{stb}\}$*:*

- $\mathbf{ABF} \vdash^{\cup}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \Delta \vdash_{\mathcal{R}} A$ *for some* $\Delta \in \mathsf{sem}(\mathbf{ABF})$.
- $\mathbf{ABF} \vdash^{\cap}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \Delta \vdash_{\mathcal{R}} A$ *for every* $\Delta \in \mathsf{sem}(\mathbf{ABF})$.
- $\mathbf{ABF} \vdash^{\Cap}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \bigcap\{\Delta \mid \Delta \in \mathsf{sem}\} \vdash_{\mathcal{R}} A$.

5 ASPIC$^+$

In ASPIC$^+$, as in ABA, inferences made on the basis of a strict knowledge base can be extended with additional inferences based on plausible assumptions. However, whereas in ABA attacks and extensions where defined directly on the level of these assumptions, in ASPIC$^+$, arguments are specific deductions. More precisely, arguments are constructed

[4] Note that $\overline{}$ does *not* denote the set theoretic complement.

[5] The former are sometimes referred to as *rule-based* and the latter as *logic-based* systems of argumentation.

[6] Some formulations of ABA define attacks on the level of individual arguments. However, since attacks are only possible 'on' assumptions, these formulations are equivalent (cf. also [20]).

from a knowledge base using an argumentation system. An argumentation system is a generalisation of a deductive system (Def. 7) that allows for a distinction between strict (i.e. deductive or safe) and defeasible rules.[7]

Definition 14 (Defeasible Theory)**.** *Given a formal language $\mathcal{L}$, a* defeasible theory $\mathsf{R} = (\mathcal{L}, \mathcal{S}, \mathcal{D})$ *consists of (where $A_1, \ldots, A_n, B \in \mathcal{L}$):*

- *a set of strict rules $\mathcal{S}$ of the form $A_1, \ldots, A_n \to B$*
- *a set of defeasible rules $\mathcal{D}$ of the form $A_1, \ldots, A_n \Rightarrow B$.*

We also assume there is a naming function $N : \mathcal{S} \cup \mathcal{D} \to \mathcal{L}$ s.t. every rule $r \in \mathcal{S} \cup \mathcal{D}$ gets assigned a unique name. $A_1, \ldots, A_n$ are called the antecedents and B is called the consequent of $A_1, \ldots, A_n \to B$ resp. $A_1, \ldots, A_n \Rightarrow B$.

Definition 15 (Argumentation System)**.** *Given a defeasible theory* R*, an* argumentation system *is a tuple $AS = (\mathsf{R}, \overline{})$ where $\overline{}$ is a contrariness function from $\mathcal{L}$ to $2^{\mathcal{L}}$.*

Arguments are built by using defeasible and/or strict rules to derive conclusions from a knowledge base. A knowledge base consists of strict and plausible premises. $\mathcal{K}_n$ is the set of all (necessary) axioms, i.e. premises that are considered to be outside the reach of argumentative attacks. $\mathcal{K}_a$ has an analogous function to the defeasible assumptions in ABA: they are deemed *plausible* in that they are assumed to be true unless and until a counterargument is encountered.

Definition 16 (Knowledge Base)**.** *A* Knowledge Base *is a set $\mathcal{K}$, where $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_a$ and $\mathcal{K}_n \cap \mathcal{K}_a = \emptyset$.*

Definition 17 (Arguments)**.** *Let $AS = (\mathsf{R}, \overline{})$ be an argumentation system and $\mathcal{K} = \mathcal{K}_a \cup \mathcal{K}_n$ a knowledge base. An* argument *a is one of the following:*

- *a* premise argument *$\langle A \rangle$ if $A \in \mathcal{K}$*
- *a* strict rule-argument *$\langle a_1, \ldots a_n \mapsto B \rangle$ if $a_1, \ldots a_n$ (with $n \geqslant 0$) are arguments such that there exists a strict rule* $\mathrm{conc}(a_1), \ldots \mathrm{conc}(a_n) \to B \in \mathcal{S}$.
- *a* defeasible rule-argument *$\langle a_1, \ldots a_n \Rightarrow B \rangle$ if $a_1, \ldots a_n$ (with $n \geqslant 0$) are arguments such that there exists a defeasible rule* $\mathrm{conc}(a_1), \ldots \mathrm{conc}(a_n) \Rightarrow B$.

We will use $\mathrm{Arg}(AS, \mathcal{K})$ to denote the set of all arguments that can be built from a knowledge base $\mathcal{K}$ using an argumentation system AS.

Example 3. *Let $\mathcal{S} = \{\neg q \to \neg p\}$, $\mathcal{D} = \{\neg p \Rightarrow s\}$, $\mathcal{K}_n = \{\neg s\}$, and $\mathcal{K}_a = \{\neg q, \neg p, q\}$. We have, e.g., the following arguments:*

$$\begin{array}{lll} a_1 = \langle \neg q \rangle & a_4 = \langle a_3 \Rightarrow s \rangle & a_7 = \langle \neg s \rangle \\ a_2 = \langle \neg p \rangle & a_5 = \langle a_2 \Rightarrow s \rangle & \\ a_3 = \langle a_1 \mapsto \neg p \rangle & a_6 = \langle q \rangle & \end{array}$$

Definition 18. *Where a is an argument $a = \langle B \rangle$, $a = \langle a_1, \ldots a_n \mapsto B \rangle$ or $a = \langle a_1, \ldots a_n \Rightarrow B \rangle$, we define:*

- $\mathrm{conc}(a) = B$
- $\mathrm{sub}(a) = \mathrm{sub}(a_1) \cup \ldots \cup \mathrm{sub}(a_n) \cup \{a\}$
- *where a is a premise argument:* $\mathrm{prem}(a) = \{A\}$
- *where a is not a premise argument:* $\mathrm{prem}(a) = \{\mathrm{prem}(a') \mid a' \in \mathrm{sub}(a)\}$.

The distinction between strict and defeasible rule-arguments allows us to define a variety of attack forms:

Definition 19 (Attacks)**.** *Where $a, b \in \mathrm{Arg}(AS, \mathcal{K})$, a attacks b (in signs, $a \rightsquigarrow b$) iff*

- $\mathrm{conc}(a) \in \overline{B}$ *for some* $B \in \mathrm{prem}(b) \cap \mathcal{K}_a$ *(*Undermining*).*
- $\mathrm{conc}(a) \in \overline{B'}$ *for some* $b' \in \mathrm{sub}(b)$ *such that* $\mathrm{conc}(b') = B'$ *and b' is of the form $\langle b'_1, \ldots, b'_n \Rightarrow B' \rangle$ (*Rebut*).*
- $\mathrm{conc}(a) = \overline{b'}$ *for some* $b' \in \mathrm{sub}(b)$ *such that b' is a defeasible argument (*Undercut*).*

Example 4 (Ex. 1, contd)**.** *Where $\overline{A} = \{B \mid B \equiv \neg A\}$ for every $A \in \mathcal{L}$, we have: $a_1 \rightsquigarrow a_6$, $a_6 \rightsquigarrow a_1$, $a_6 \rightsquigarrow a_3$, $a_6 \rightsquigarrow a_4$, $a_7 \rightsquigarrow a_4$, $a_7 \rightsquigarrow a_5$.*

Definition 20 (Structured Argumentation Framework)**.** *A* structured argumentation framework *$\mathbf{AT} = (\mathrm{Arg}(AS, \mathcal{K}), \rightsquigarrow)$ is a pair where $\mathrm{Arg}(AS, \mathcal{K})$ is the set of arguments built from $\mathcal{K}$ using the argumentation system AS and $\rightsquigarrow$ is an attack relation over $\mathrm{Arg}(AS, \mathcal{K})$.*

Given a structured argumentation framework, we can again make use of Dung's argumentation semantics to define different notions of acceptable sets of arguments.

Definition 21 (Argumentation Semantics)**.** *Given a structured argumentation framework $\mathbf{AT} = (Arg(AS, \mathcal{K}), \rightsquigarrow)$, where $\mathcal{B} \subseteq Arg(AS, \mathcal{K})$,*

- *$\mathcal{B}$ is* conflict-free *iff there is no $a, b \in \mathcal{B}$ such that $a \rightsquigarrow b$*
- *$\mathcal{B}$ is* naive *iff it is maximally conflict-free.*
- *$\mathcal{B}$* defends *$a \in \mathcal{A}$ iff for every $c \in \mathcal{A}$ for which $c \rightsquigarrow a$, there is a $b \in \mathcal{B}$ such that $b \rightsquigarrow c$.*
- *$\mathcal{B}$ is* admissible *iff it is conflict-free and it defends every argument $a \in \mathcal{B}$*
- *$\mathcal{B}$ is* preferred *iff it is maximally (w.r.t. set inclusion) admissible.*
- *$\mathcal{B}$ is* stable *iff it is conflict-free and for every $a \in Arg(AS, \mathcal{K}) \setminus \mathcal{B}$, $\mathcal{B} \rightsquigarrow a$.*

We write $\mathsf{niv}(\mathbf{AT}), \mathsf{prf}(\mathbf{AT})$ resp. $\mathsf{stb}(\mathbf{AT})$ for the set of naive, preferred resp. stable sets of arguments in $\mathbf{AT}$.

Definition 22. *Where $\mathbf{AT} = (Arg(AS, \mathcal{K}), \rightsquigarrow)$ is a structured argumentation framework and $\mathsf{sem} \in \{\mathsf{niv}, \mathsf{prf}, \mathsf{stb}\}$,*

- $\mathbf{AT} \vdash^{\cup}_{\mathsf{sem}} A$ *iff there is an $a \in \mathcal{B}$ with* $\mathrm{conc}(a) = A$ *for some $\mathcal{B} \in \mathsf{sem}(\mathbf{AT})$.*
- $\mathbf{AT} \vdash^{\cap}_{\mathsf{sem}} A$ *iff for every $\mathcal{B} \in \mathsf{sem}(\mathbf{AT})$ there is an $a \in \mathcal{B}$ with* $\mathrm{conc}(a) = A$.
- $\mathbf{AT} \vdash^{\Cap}_{\mathsf{sem}} A$ *iff there is an $a \in \mathcal{B}$ with* $\mathrm{conc}(a) = A$ *for every $\mathcal{B} \in \mathsf{sem}(\mathbf{AT})$.*

[7] In the ASPIC$^+$ framework of [16], there is also the possibility to add a preference ordering over the premises and/or defeasible rules. Similar generalisations exist for ALs and approaches based on maximal consistent subsets and their generalisations such as Makinsons' default assumptions. We will present investigations into translations for systems with priorities at a future occasion. In our presentation, we also disregard a special type of premise called 'issue' in the context of ASPIC$^+$. Issues are premises that are never acceptable in the sense that they always require further backup by additional arguments.

6 Translating Adaptive Logic to Assumption-Based Argumentation

The idea of the translation from ALs to ABA is the following. We translate the lower limit logic $\mathbf{L}$ of the given AL into a deductive system, plausible assumptions are negations of abnormalities, and the contrariness operator is classical negation. Recall that the lower limit logic is a supraclassical Tarski logic. Hence, there are classical negation $\neg$ and classical disjunction $\vee$ in the underlying language of $\mathbf{L}$. In the remainder of this section we will use $\neg$ and $\vee$ denoting these classical connectives.

We now go through the technical details of our translation.

Definition 23. *Let* $\mathbf{AL}$ *be an AL with the lower limit logic* $\mathbf{L}$ *in a formal language* $\mathcal{L}$ *and the consequence relation* $\vdash_{\mathbf{L}}$*, the set of abnormalities* $\Omega \subseteq \mathcal{L}$ *and a strategy* str *(reliability, minimal abnormality, or normal selections). Let* $\mathbf{L}$ *be characterised by the rules in* R *and the axiom schemes in* A*. We the define the assumption based framework* $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$ *for the premise set* $\Gamma \subseteq \mathcal{L}$ *as the tuple* $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma) = ((\mathcal{L}, \mathcal{R}(\mathbf{L})), \Gamma, Ab_{\Omega}, \overline{})$ *where:*

- $\mathcal{R}(\mathbf{L})$ *contains all instances of rules in* R *and a rule* $\rightarrow A$ *for all instances* A *of axiom schemes in* A*;*[8]
- $Ab_{\Omega} = \{\neg A \mid A \in \Omega\}$
- $\overline{} : Ab_{\Omega} \rightarrow \mathcal{L}$*, where* $\overline{\neg A} = A$

Below we show the following representational theorem:

Theorem 6. *Where* $\Gamma \cup \{A\} \subseteq \mathcal{L}$ *and* $\mathsf{sem} \in \{\mathsf{niv}, \mathsf{prf}, \mathsf{stb}\}$,

1. $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma) \vdash^{\cup}_{\mathsf{sem}} A$ *iff* $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{ns}} A$

2. $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma) \vdash^{\cap}_{\mathsf{sem}} A$ *iff* $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{ma}} A$

3. $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma) \vdash^{\Cap}_{\mathsf{sem}} A$ *iff* $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{r}} A$.

To avoid clutter we introduce some notational convention:

Notation 1. *Where* $\Delta \subseteq \Omega$, $\Delta^{\neg} = \{\neg A \mid A \in \Delta\}$ *and* $\overline{\Delta^{\neg}} = \Delta$.

The following fact follows immediately in view of the compactness and the transitivity of $\mathbf{L}$.

Fact 2. *Where* $\Gamma \cup \{A\} \subseteq \mathcal{L}$, $\Gamma \vdash_{\mathcal{R}(\mathbf{L})} A$ *iff* $\Gamma \vdash_{\mathbf{L}} A$.

In view of this fact, we will indiscriminately use $\vdash$ as $\vdash_{\mathcal{R}(\mathbf{L})}$ and $\vdash_{\mathbf{L}}$. Note that in view of the supraclassicality of $\mathbf{L}$ we have:

Fact 3. $\Gamma \cup \Delta^{\neg} \vdash A$ *iff* $\Gamma \vdash \bigvee \overline{\Delta^{\neg}} \vee A$.

We now established that every instantiation of an AL is indeed an assumption-based framework. We prove that the three consequence relations of ALs correspond to intuitive ways of calculating consequences in ABA. The crucial result to prove this is the fact that every preferred extension in some assumption-based framework $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$ is exactly the set of negations of abnormalities excluding some choice set over the derivable abnormalities. This is shown in the following lemmas.

Lemma 1. *Where* $\phi \in \Phi_{\mathbf{L}}(\Gamma)$*,* $Ab_{\Omega} \setminus \phi^{\neg}$ *is stable in* $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$.

Proof. We first show that $\Delta^{\neg} = Ab_{\Omega} \setminus \phi^{\neg}$ is conflict-free. Assume for a contradiction that it is not and hence that there is a $B \in \Omega$ for which $\Gamma \cup \Delta^{\neg} \vdash B, \neg B$. Hence, by the compactness of $\mathbf{L}$ and Fact 3, $\Gamma \vdash \bigvee \Theta$ for some finite $\Theta \subseteq \Delta$. Let Θ be $\subset$-minimal with this property. Hence, $\Theta \in \Sigma_{\mathbf{L}}(\Gamma)$. However, then $\phi \cap \Theta \neq \emptyset$, a contradiction.

We now show that $\Delta^{\neg}$ is stable. For this, let $\neg B \in Ab_{\Omega} \setminus \Delta^{\neg}$. Hence, $B \in \phi$. With Fact 1.2, there is a $\Theta \in \Sigma_{\mathbf{L}}(\Gamma)$ such that $\{B\} = \phi \cap \Theta$. Since $\Gamma \vdash \bigvee \Theta$, by Fact 3 also $\Gamma \cup (\Theta^{\neg} \setminus \{\neg B\}) \vdash B$. By the monotonicity of $\mathbf{L}$, $\Gamma \cup \Delta^{\neg} \vdash B$ which means that Δ attacks B.

Since $\Delta^{\neg}$ is conflict-free and attacks every $A \in Ab_{\Omega} \setminus \Delta^{\neg}$, it is easy to see that $\Delta^{\neg}$ is closed and stable. □

Example 5 (Ex. 1 contd)**.** *Take* $Ab_{\Omega} = \{\neg(A \wedge {\sim}A) \mid A \in \mathcal{L}_{\mathbf{CLuN}}\}$ *and* $\mathcal{R}$ *an adequate rule system for* $\mathbf{CLuN}$*. Where* $\Gamma = \{{\sim}p, {\sim}q, p \vee q, p \vee r, q \vee s\}$*. There are two stable extensions:* $Ab_{\Omega} \setminus \{\neg(p \wedge {\sim}p)\}$ *and* $Ab_{\Omega} \setminus \{\neg(q \wedge {\sim}q)\}$*. To see this observe that e.g.* $\Gamma \cup \{\neg(q \wedge {\sim}q)\} \vdash_{\mathbf{CLuN}} p \wedge {\sim}p$*.*

Lemma 2. *If* $\Delta^{\neg} \subseteq Ab_{\Omega}$ *is conflict-free in* $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$ *then there is a* $\phi \in \Phi_{\mathbf{L}}(\Gamma)$ *for which* $\Delta \subseteq \Omega \setminus \phi$*.*

Proof. Suppose $\Delta \not\subseteq \Omega \setminus \phi$ for all $\phi \in \Phi_{\mathbf{L}}(\Gamma)$ and $\Delta \subseteq \Omega$. By Fact 1, $\Omega \setminus \Delta$ is not a choice set of $\Sigma_{\mathbf{L}}(\Gamma)$. Thus, there is a $\Theta \in \Sigma_{\mathbf{L}}(\Gamma)$ for which $\Theta \subseteq \Delta$. Since $\Gamma \vdash \bigvee \Theta$, also $\Gamma \cup (\Theta \setminus \{A\}) \vdash \neg A$ for any $A \in \Theta$. Thus, $\Gamma \cup \Delta$ is not $\mathbf{L}$-consistent since $\Gamma \cup \Delta \vdash A, \neg A$ by monotonicity. By Fact 2, $\Gamma \cup \Delta \vdash_{\mathcal{R}(\mathbf{L})} A, \neg A$ and thus, Δ is not conflict-free in $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$. □

With Lemmas 1 and 2 we immediately get:

Lemma 3. *Where* $\Gamma \subseteq \mathcal{L}$, $\{Ab_{\Omega} \setminus \phi^{\neg} \mid \phi \in \Phi_{\mathbf{L}}(\Gamma)\} = \mathsf{stb}(\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)) = \mathsf{prf}(\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)) = \mathsf{niv}(\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma))$

We are now in a position to prove our main result in this section:

Proof of Theorem 6. In view of Lemma 3 it is enough to show the theorem for $\mathsf{sem} = \mathsf{stb}$.

Ad 3. $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma) \vdash^{\Cap}_{\mathsf{stb}} A$ iff $\Gamma \cup \bigcap\{\Delta \mid \Delta \in \mathsf{stb}(\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma))\} \vdash A$. By Lemma 1, this is the case iff $\Gamma \cup \bigcap\{(\Omega \setminus \phi)^{\neg} \mid \phi \in \Phi_{\mathbf{L}}(\Gamma)\} \vdash A$. Since $\bigcup \Phi_{\mathbf{L}}(\Gamma) = \bigcup \Sigma_{\mathbf{L}}(\Gamma)$ (which is easy to see and left to the reader), this is equivalent to $\Gamma \cup (\Omega \setminus \bigcup \Sigma_{\mathbf{L}}(\Gamma))^{\neg} \vdash A$. By compactness, monotonicity and Fact 3, this is equivalent to $\Gamma \vdash A \vee \bigvee \Delta$ for some finite $\Delta \subseteq \Omega \setminus \bigcup \Sigma_{\mathbf{L}}(\Gamma)$. By Theorem 2 this is equivalent to $\Gamma \vdash^{\Omega,\mathbf{L}}_{\mathsf{r}} A$.

Ad 1. and 2. Analogous. □

Translating Adaptive Logic to ASPIC$^+$

In [16] we have a translation from ABA to ASPIC$^+$. Although this translation requires several assumptions that $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$ does not satisfy, it turns out that it is easy to prove that any $\mathbf{ABF}^{\Omega}_{\mathbf{L}}(\Gamma)$ can easily be translated to an assumption-based framework that does satisfy these assumptions.

The underlying idea is basically the same as that for translating AL into ABA: the plausible knowledge base consists of the negated abnormalities, the strict premises of the ASPIC$^+$ framework are the premise set Γ and the strict rules

[8] If no axiomatisation of $\mathbf{L}$ is given, we can proceed more brute force and set $\mathcal{R} = \{A_1, \ldots, A_n \rightarrow A \mid \{A_1, \ldots, A_n\} \vdash_{\mathbf{L}} A\}$.

of the ASPIC$^+$ framework are the inference rules of the monotonic core logic. Due to spatial restrictions, we are not able to present the full technical details of this translation and the adequacy results here.

7 Translating ASPIC$^+$ to Assumption-Based Argumentation

In this section we translate ASPIC$^+$ to ABA. Since in ABA we have no defeasible rules and less attack types than in ASPIC$^+$ the possibility of this translation is less expected than the translation in the other direction (as provided in [16]). In this section we thus offer an answer to the open question stated in [14] whether such a translation can be given. Our translation works as follows:

Definition 24. *Where* $\mathrm{AS} = (\mathsf{R}, \overline{})$ *is an argumentation system in the formal language* $\mathcal{L}$ *with a naming function* N *for the rules in* R *and* $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_a$ *is a knowledge base, we translate* AS *into an assumption-based framework* $\mathbf{ABF}(\mathrm{AS}) = ((\mathcal{L}', \mathcal{R}), \mathcal{K}_n, Ab, \overline{})$ *as follows*[9]

- $\mathcal{L}' \supseteq \mathcal{L}$ *is such that* $\mathcal{L}' \setminus \mathcal{L}$ *contains for each* r *in* R *a unique name* $n(r)$ *and its contrary* $\overline{n(r)}$*;*[10]
- $\mathcal{R}$ *contains each strict rule from* R *and for each defeasible rule* $r : A_1, \ldots, A_n \Rightarrow A$ *it contains*[11]
  - *the rule* $n(r), A_1, \ldots, A_n \to A$
  - *the rule* $\overline{A} \to \overline{n(r)}$
- $Ab = \mathcal{K}_a \cup \{n(r) \mid r \textit{ is a defeasible rule in } \mathsf{R}\}$

Below we will show that the translation is adequate in view of the following corollary:

Corollary 1. *Where* $\mathbf{AT} = (\mathrm{Arg}(\mathrm{AS}, \mathcal{K}), \rightsquigarrow)$ *is a structured argumentation framework and* $\mathsf{sem} \in \{\mathsf{stb}, \mathsf{prf}\}$*,*

1. $\mathbf{ABF}(\mathrm{AS}) \vdash^{\cup}_{\mathsf{sem}} A$ *iff* $\mathbf{AT} \vdash^{\cup}_{\mathsf{sem}} A$
2. $\mathbf{ABF}(\mathrm{AS}) \vdash^{\cap}_{\mathsf{sem}} A$ *iff* $\mathbf{AT} \vdash^{\cap}_{\mathsf{sem}} A$*.*
3. $\mathbf{ABF}(\mathrm{AS}) \vdash^{\Cap}_{\mathsf{sem}} A$ *iff* $\mathbf{AT} \vdash^{\Cap}_{\mathsf{sem}} A$*.*

In the following we suppose a given argumentation system AS and its translation $\mathbf{ABF}(\mathrm{AS})$ as in Definition 24.

Definition 25. *Where* $\Delta \subseteq Ab$*,* $\mathrm{Arg}_\Delta \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$ *is the set of all arguments* a *that use only defeasible assumptions in* Δ*, any strict rules, and only defeasible rules* r *for which* $n(r) \in \Delta$.

Where $\mathcal{A} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$ *is a set of arguments,* $Ab_{\mathcal{A}} \subseteq Ab$ *is the set of assumptions consisting of (1) defeasible assumptions* $A \in \mathcal{K}_a$ *for which* $\mathrm{prem}(a) = A$ *or* $\mathrm{conc}(a) = A$ *for some* $a \in \mathcal{A}$ *and (2) of* $n(r)$ *where* r *is a defeasible rule used in some argument in* $\mathcal{A}$*.*

Where $\mathcal{A}$ *is a set of arguments in* $\mathrm{Arg}(\mathrm{AS}, \mathcal{K})$*,* $\mathcal{A}^\star$ *denotes the set* $\mathrm{Arg}_{Ab_{\mathcal{A}}}$.

We sometimes write Ab_a instead of $Ab_{\{a\}}$.

Fact 4. *Where* $\mathcal{A} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$ *is a set of assumptions,* $\mathcal{A} \subseteq \mathcal{A}^\star$*.*

Lemma 4. *Where* $A \neq n(r)$ *for any* r *in* R *and* $\Delta \subseteq Ab$*, if* $\mathcal{K}_n \cup \Delta \vdash_{\mathcal{R}} A$ *then*

1. *if* $A \in \mathcal{L}$*, there is an* $a \in \mathrm{Arg}_\Delta$ *such that* $\mathrm{conc}(a) = A$*,*
2. *else (if* $A = \overline{n(r)}$*), there is an* $a \in \mathrm{Arg}_\Delta$ *for which* $\mathrm{conc}(a) = \overline{B}$ *where* B *is the consequent of* r*.*

Proof. This can be shown by an induction on the length of a deduction from $\mathcal{K}_n \cup \Delta$ to A. Base step: this is trivial since $A \in \mathcal{K}$. Inductive step. We have three possibilities:

1. A is the result of applying a strict rule r in R to $A_1, \ldots, A_n$, or
2. A is the result of applying the translation of a defeasible rule $r = A_1, \ldots, A_n \Rightarrow A \in \mathsf{R}$ to $A_1, \ldots, A_n$ and the rule name $n(r)$, or
3. $A = \overline{n(r)}$ is the result of applying a rule $\overline{B} \to \overline{n(r)}$ where B is the consequent of the defeasible rule r in R.

Ad 2. By the induction hypothesis there are arguments a_i $(1 \leq i \leq n)$ s.t. $a_i \in \mathrm{Arg}_\Delta$ and $\mathrm{conc}(a_i) = A_i$. (Note here that $A_i \notin \mathcal{L}' \setminus \mathcal{L}$.) Clearly, $a = \langle a_1, \ldots, a_n \Rightarrow A \rangle \in \mathrm{Arg}_\Delta$ since $n(r) \in \Delta$. Ad 1. Analogous. Ad 3. By the induction hypothesis and since $\overline{B} \in \mathcal{L}$, there is an argument $a \in \mathrm{Arg}_\Delta$ with $\mathrm{conc}(a) = \overline{B}$. □

The other direction of Lemma 4.1 follows immediately in view of Definition 25:

Fact 5. *Where* $\mathcal{A} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$*, if there is an* $a \in \mathcal{A}$ *with* $\mathrm{conc}(a) = A$ *then* $\mathcal{K}_n \cup Ab_{\mathcal{A}} \vdash_{\mathcal{R}} A$*.*

Lemma 5. *Where* $\mathcal{A} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$*, if* $\mathcal{A}$ *is admissible then* $\mathcal{A}^\star$ *is admissible.*

Proof. Suppose there are a and $b \in \mathcal{A}^\star$ s.t. a attacks b. For each attack form it is easy to see that then there is a $b'' \in \mathcal{A}$ s.t. a attacks b''. Take, for instance, rebuttal. Then $\mathrm{conc}(a) = \overline{B}'$ where $B' = \mathrm{conc}(b')$ for some $b' \in \mathrm{sub}(b)$. Hence, there is a defeasible rule r which is applied in b' to produce B'. By the definition of $\mathrm{Arg}_{Ab_{\mathcal{A}}}$there is an argument $b'' \in \mathcal{A}$ s.t. r is applied to produce $\mathrm{conc}(b'') = B'$. For the other attack types (undercuts and undermines) this is shown in an analogous way. Now, since $\mathcal{A}$ is admissible, there is a $c \in \mathcal{A}$ s.t. c attacks a. Since by Fact 4, $c \in \mathcal{A}^\star$, also $\mathcal{A}^\star$ is defended. To show that $\mathcal{A}^\star$ is conflict-free, assume for a contradiction that $a \in \mathcal{A}^\star$. Since a attacks $b'' \in \mathcal{A}$, $\mathcal{A}$ attacks a (due to the admissibility of $\mathcal{A}$). However, in view of the fact that $\mathcal{A}$ and $\mathcal{A}^\star$ make use of the same defeasible assumptions and defeasible rules and $\mathcal{A}$ attacks a in one of the two, this leads to a selfattack in some argument $a' \in \mathcal{A}$. E.g., suppose $\mathcal{A}$ undermines a in some $B \in \mathrm{prem}(a)$. Then $B \in Ab_{\mathcal{A}}$. Hence there is an argument $a' \in \mathcal{A}$ with $B \in \mathrm{prem}(a')$ and $\mathcal{A}$ attacks a'. Since $\mathcal{A}$ is conflict-free, this is a contradiction. □

[9] For simplicity, we will assume that the contrariness function of the ASPIC$^+$-framework assigns a unique contrary to every $A \in \mathcal{L}$. If this assumption is not satisfied, one has to add $A_1^c \to -A, \ldots, A_n^c \to -A$ for every $A_i^c \in \bar{A}$, where $-A \in \mathcal{L}' \setminus \mathcal{L}$ is the contrary of A in ABA, as suggested by [20, p.109].

[10] Formally: $\mathcal{L}' \setminus \mathcal{L} = \{n(r) \mid r \text{ in } \mathsf{R}\} \cup \{\overline{n(r)} \mid r \text{ in } \mathsf{R}\}$ (where $\{n(r) \mid r \text{ in } \mathsf{R}\} \cap \{\overline{n(r)} \mid r \text{ in } \mathsf{R}\} = \emptyset$). This warrants that, unlike the names $N(r) \in \mathcal{L}$ used in AS, the new names $n(r)$ are not antecedents and consequents of rules in R. We use the new names to 'simulate' defeasible rules in ABA.

[11] We suppose that the rules in $\mathcal{R}$ are instances as opposed to schemes. The translation can easily be adjusted to schemes.

Lemma 6. *Where $\mathcal{A} = \mathcal{A}^{\star} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$, $Ab_{\mathcal{A}}$ is closed.*

Proof. Suppose $\mathcal{A} = \mathcal{A}^{\star}$ and $\mathcal{K}_n \cup Ab_{\mathcal{A}} \vdash_{\mathcal{R}} A$ for some $A \in Ab$. We have two possibilities: (1) $A = n(r)$ for some r in R or (2) $A \in \mathcal{K}_a$. Ad 1. Since there are no rules with consequent $n(r)$, $n(r) \in Ab_{\mathcal{A}}$. Ad 2. By Lemma 4, there is an $a \in \mathcal{A}^{\star} = \mathcal{A}$ with $\mathrm{conc}(a) = A$. Hence, by the definition of $Ab_{\mathcal{A}}$, $A \in Ab_{\mathcal{A}}$. □

Lemma 7. *Where $\mathcal{A} = \mathcal{A}^{\star} \subseteq \mathrm{Arg}(\mathrm{AS}, \mathcal{K})$, if $\mathcal{A}$ is admissible then $Ab_{\mathcal{A}}$ is admissible.*

Proof. Suppose $\mathcal{A} = \mathcal{A}^{\star}$. By Lemma 6, $Ab_{\mathcal{A}}$ is closed. Suppose $Ab_{\mathcal{A}}$ is not conflict-free. Hence, $\mathcal{K}_n \cup Ab_{\mathcal{A}} \vdash_{\mathcal{R}} \overline{A}$ for some $A \in Ab_{\mathcal{A}}$. We use Lemma 4 according to which we have two cases. Case 1: there is an $a \in \mathcal{A}^{\star}$ s.t. $\mathrm{conc}(a) = \overline{A}$. Since $\mathcal{A} = \mathcal{A}^{\star}$, $a \in \mathcal{A}$ and $\mathcal{A}$ is not conflict-free. Case 2: $A = n(r)$ and there is an $a \in \mathcal{A}$ for which $\mathrm{conc}(a) = \overline{B}$ where B is the consequent of r. Since $n(r) \in Ab_{\mathcal{A}}$, there is an argument $a' \in \mathcal{A}$ which uses rule r to produce $\mathrm{conc}(a') = B$ and which is thus rebut-attacked by a. Again, $\mathcal{A}$ is not conflict-free. Thus, we have shown (by contraposition) that if $\mathcal{A}$ is conflict-free then $Ab_{\mathcal{A}}$ is conflict-free.

Suppose $\mathcal{A}$ is admissible, Δ is closed and attacks $Ab_{\mathcal{A}}$. Hence, $\mathcal{K}_n \cup \Delta \vdash_{\mathcal{R}} \overline{A}$ for some $A \in Ab_{\mathcal{A}}$. By Lemma 4 we have two cases. Case 1: there is an $a \in \mathrm{Arg}_{\Delta}$ s.t. $\mathrm{conc}(a) = \overline{A}$. Hence, $A \neq n(r)$ for any $r \in$ R. Clearly, a attacks $\mathcal{A}$. Since $\mathcal{A}$ is admissible, there is a $b \in \mathcal{A}$ s.t. b attacks a. Then $Ab_b \subseteq Ab_{\mathcal{A}}$ and $\mathcal{K}_n \cup Ab_b \vdash_{\mathcal{R}} \mathrm{conc}(b)$. Thus, Ab_b attacks Ab_a and hence $Ab_{\mathcal{A}}$ attacks Δ.

Case 2: $A = n(r)$ and there is an $a \in \mathrm{Arg}_{\Delta}$ s.t. $\mathrm{conc}(a) = \overline{B}$ where B is the consequent of r. In this case there is an $a' \in \mathcal{A}$ which uses rule r and hence $\mathrm{conc}(a') = B$. Since $\mathcal{A}$ is admissible, there is a $c \in \mathcal{A}$ that attacks a. But then $\Delta_c \subseteq Ab_{\mathcal{A}}$ attacks Δ_a and hence $Ab_{\mathcal{A}}$ attacks Δ. □

Lemma 8. *If $\Delta \subseteq Ab$ is admissible, then Arg_{Δ} is admissible.*

Proof. Similar to the previous proof. □

Theorem 7. *1. If Δ is preferred (resp. stable) then Arg_{Δ} is preferred (resp. stable).*

2. If $\mathcal{A}$ is preferred (resp. stable) then Δ is preferred (resp. stable) for some $\Delta \supseteq Ab_{\mathcal{A}}$ for which $\mathrm{Arg}_{\Delta} = \mathcal{A}$.

Proof. Ad.1 Suppose Δ is preferred. Then, by Lemma 8, Arg_{Δ} is admissible. Suppose there is an $\mathcal{A}' \supset \mathrm{Arg}_{\Delta}$ that is admissible, then by Lemma 7, also $Ab_{\mathcal{A}'}$ is admissible. Since $\Delta \subset Ab_{\mathcal{A}'}$ this is a contradiction.

Ad.2 Suppose $\mathcal{A}$ is preferred. By Lemma 5 and since trivially $\mathcal{A} \subseteq \mathcal{A}^{\star}$, $\mathcal{A} = \mathcal{A}^{\star}$. By Lemma 7, $Ab_{\mathcal{A}}$ is admissible. Now suppose that there is a $\Delta \supset Ab_{\mathcal{A}}$ that is admissible. Then by Lemma 8, Arg_{Δ} is admissible. Clearly $\mathcal{A} \subseteq \mathrm{Arg}_{\Delta}$. By the maximality of $\mathcal{A}$, $\mathcal{A} = \mathrm{Arg}_{\Delta}$.

Due to space limitations we omit the proof for stable extensions. □

Corollary 1 follows directly with Theorem 7, Lemma 4 and Fact 5.

8 Translating Assumption-based Argumentation to Adaptive Logic

In this section we will translate a fragment of assumption-based argumentation to adaptive logic.

In the following we write $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$for the assumption-based framework $((\mathcal{L}, \mathcal{R}), \Gamma, Ab, \overline{})$.

For our translation we will use some connectives from Kleene's well-known 3-valued logic $\mathbf{K}_3$ (see Table 1) and superimpose them on a logic that is characterised by the rules in $\mathcal{R}$. This works as follows.

We define the 3-valued logic $\mathbf{L}^3_{\mathcal{R}}$ semantically in the following way: we superimpose on the language $\mathcal{L}$ the operators $\sim$ and $\vee$ (which are supposed to not occur in the alphabet of $\mathcal{L}$) resulting in the set of well-formed formulas $\mathcal{L}^3_{\mathcal{R}}$. The operators are characterised by the truth tables in Table 1.[12]

A	$\overline{A}$
1	0
0	1
u	u

A	$\sim A$
1	0
0	1
u	1

$\vee$	1	0	u
1	1	1	1
0	1	0	u
u	1	u	u

Table 1: Truth-tables for $\overline{}$, $\sim$ and $\vee$.

Definition 26. *$v : \mathcal{L} \to \{0, 1, u\}$ is a function which respects the truth-table for $\overline{}$ (i.e., $v(\overline{A}) = 1$ iff $v(A) = 0$, $v(\overline{A}) = 0$ iff $v(A) = 1$, and $v(\overline{A}) = u$ iff $v(A) = u$). The valuation function $v_M : \mathcal{L}^3_{\mathcal{R}} \to \{0, u, 1\}$ is defined inductively as follows:*

1. *where $A \in \mathcal{L}$, $v_M(A) = v(A)$*
2. *$v_M(\sim A) = 0$ iff $v_M(A) = 1$, and $v_M(\sim A) = 1$ else.*
3. *$v_M(A \vee B) = \max(v_M(A), v_M(B))$ where $0 < u < 1$.*

We write $M \models A$ iff $v_M(A) = 1$ (so 1 is the only designated value). We write $\Vdash_{\mathbf{L}^3_{\mathcal{R}}}$ for the resulting consequence relation.

We now use $\mathbf{L}^3_{\mathcal{R}}$ as a lower limit logic for an adaptive logic with the set of abnormalties:

Notation 2. $\Omega^{\sim}_{Ab} = \{\sim A \mid A \in Ab\}$.

We translate the rules of $\mathcal{R}$ as follows: $A_1, \ldots, A_n \to B$ is translated to $\sim A_1 \vee \ldots \vee \sim A_n \vee B$.

Notation 3. *Where $\mathcal{R}$ is a set of rules, we write $\mathcal{R}^{\sim}$ for the set of translated rules.*

Our two main representational results in this section are (to be proven below):

Theorem 8. *Where $\Gamma \cup \{A\} \subseteq \mathcal{L}$, and $\mathsf{sem} = \mathsf{niv}$,*

1. $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma) \vdash^{\cup}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \mathcal{R}^{\sim} \Vdash^{\Omega^{\sim}_{Ab}, \mathbf{L}^3_{\mathcal{R}}}_{\mathsf{ns}} A$
2. $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma) \vdash^{\cap}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \mathcal{R}^{\sim} \Vdash^{\Omega^{\sim}_{Ab}, \mathbf{L}^3_{\mathcal{R}}}_{\mathsf{ma}} A$
3. $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma) \vdash^{\Cap}_{\mathsf{sem}} A$ *iff* $\Gamma \cup \mathcal{R}^{\sim} \Vdash^{\Omega^{\sim}_{Ab}, \mathbf{L}^3_{\mathcal{R}}}_{\mathsf{r}} A$

[12] In the terminology of [21], $\overline{\text{O}}$ur negation $\sim$ corresponds to Bochvar's 'external negation' and $\overline{}$ corresponds to Kleene's negation in his $\mathbf{K}_3$. Our disjunction $\vee$ is Kleene's strong disjunction. The requirement of supraclassicality for $\mathbf{L}^3_{\mathcal{R}}$ to serve as a core logic for an AL is satisfied in view of the $\langle \vee, \sim \rangle$-fragment of $\mathbf{L}^3_{\mathcal{R}}$.

We can strengthen our result if we suppose that the rule system based on $\mathcal{R}$ satisfies the following requirement: where $\Gamma \cup \{A\} \subseteq \mathcal{L}$,

EX Where $\Delta \subseteq Ab$ is naive in $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$ and $A \in Ab \backslash \Delta$, $\Gamma \cup \Delta \vdash_{\mathcal{R}} \overline{A}$.

This criterion ensures that every naive set is stable.

Theorem 9. *Where $\Gamma \cup \{A\} \subseteq \mathcal{L}$: if $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$ satisfies (EX), items 1–3 in Theorem 8 hold for* $\mathsf{sem} \in \{\mathsf{niv}, \mathsf{prf}, \mathsf{stb}\}$.

We are now going to prove the two theorems above. The following notation will be convenient to avoid clutter:

Notation 4. $\Delta^{\sim} = \{\sim A \mid A \in \Delta\}$.

The following facts will be useful below:

Fact 6. *Where $\Gamma \cup \Delta \cup \{A\} \subseteq \mathcal{L}^3_{\mathcal{R}}$, (i) $\overline{A} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} {\sim}A$, (ii) $\Gamma \Vdash_{\mathbf{L}^3_{\mathcal{R}}} \bigvee \Delta^{\sim} \vee A$ iff $\Gamma \cup \Delta \Vdash_{\mathbf{L}^3_{\mathcal{R}}} A$.*

$\mathbf{L}^3_{\mathcal{R}}$ is obviously a compact Tarski logic.

We say that $\Gamma \subseteq \mathcal{L}$ is $\mathcal{R}$-consistent iff there is no A such that $\Gamma \vdash_{\mathcal{R}} A, \overline{A}$.

Lemma 9. *Where $\Gamma \cup \{A\} \subseteq \mathcal{L}$,*

1. $\Gamma \vdash_{\mathcal{R}} A$ implies $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} A$

2. if Γ is $\mathcal{R}$-consistent, $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} A$ implies $\Gamma \vdash_{\mathcal{R}} A$.

Proof. Ad 1. Simple induction on the number of proof steps. We show the induction step. Let M be a model of $\Gamma \cup \mathcal{R}^{\sim}$. Suppose A follows by means of the application of a rule $A_1, \ldots, A_n \to B$. By the induction hypothesis, $M \models A_1, \ldots, A_n$. Also, $M \models {\sim}A_1 \vee \ldots \vee {\sim}A_n \vee B$. Hence, with the truth-tables for $\sim$ and $\vee$, $M \models B$.

Ad 2. Suppose $\Gamma \nvdash_{\mathcal{R}} A$. We now construct a countermodel M of $\Gamma \cup \mathcal{R}^{\sim}$ for A as follows. Let

$$v : B \mapsto \begin{cases} 1 & \Gamma \vdash_{\mathcal{R}} B \\ 0 & \Gamma \vdash_{\mathcal{R}} \overline{B} \\ u & \text{else} \end{cases}$$

Note that $v(A) \in \{u, 0\}$ and hence $M \not\models A$. We have to show that M is a model of $\Gamma \cup \mathcal{R}^{\sim}$. Since Γ is $\mathcal{R}$-consistent, the definition warrants that the truth-table for $\overline{}$ is respected by v. We thus only have to check whether M verifies all formulas in $\Gamma \cup \mathcal{R}^{\sim}$. As for Γ this holds trivially since every $B \in \Gamma$ is such that $\Gamma \vdash_{\mathcal{R}} B$ and thus $v(B) = 1$. Let now $A_1, \ldots, A_n \to B \in \mathcal{R}$. We have to check whether $M \models (\bigvee_{i=1}^{n} {\sim}A_i) \vee B$. Assume the opposite. Thus $v_M(A_i) = 1$ $(1 \leq i \leq n)$ and $v_M(B) \in \{0, u\}$. But then $\Gamma \vdash_{\mathcal{R}} A_i$ $(1 \leq i \leq n)$ and thus $\Gamma \vdash_{\mathcal{R}} B$. Hence, $v_M(B) = 1$, a contradiction. □

We say that a Γ is $\mathbf{L}^3_{\mathcal{R}}$-consistent, if there is a $A \in \mathcal{L}^3_{\mathcal{R}}$ for which $\Gamma \nVdash_{\mathbf{L}^3_{\mathcal{R}}} A$.[13]

Lemma 10. *Where $\Gamma \subseteq \mathcal{L}$, if Γ is $\mathcal{R}$-consistent then $\Gamma \cup \mathcal{R}^{\sim}$ is $\mathbf{L}^3_{\mathcal{R}}$-consistent.*

Proof. Suppose Γ is $\mathcal{R}$-consistent. Then $\Gamma \nvdash_{\mathcal{R}} A, \overline{A}$ for any $A \in \mathcal{L}$. By Lemma 9, also $\Gamma \cup \mathcal{R}^{\sim} \nVdash_{\mathbf{L}^3_{\mathcal{R}}} A, \overline{A}$ for any $A \in \mathcal{L}$. □

Lemma 11. *Where $\Gamma \subseteq \mathcal{L}$: $\Delta^{\sim} \subseteq \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$, if Δ is conflict-free in $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$.*

Proof. Suppose $\Delta^{\sim} \not\subseteq \Omega^{\sim}_{Ab} \setminus \phi$ for all $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$ and $\Delta \subseteq \text{Ab}$. By Fact 1, $\Omega^{\sim}_{Ab} \setminus \Delta^{\sim}$ is not a choice set of $\Sigma_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$ which means that there is a $\Theta^{\sim} \in \Sigma_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$ such that $\Theta \subseteq \Delta$. Since $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} \bigvee \Theta^{\sim}$, by Fact 6 also $\Gamma \cup (\Theta \setminus \{A\}) \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} {\sim}A$ for any $A \in \Theta$. Hence, $\Gamma \cup \Delta \cup \mathcal{R}^{\sim}$ is not $\mathbf{L}^3_{\mathcal{R}}$-consistent since $\Gamma \cup \Delta \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} A, {\sim}A$. Thus by Lemma 10, $\Gamma \cup \Delta$ is not $\mathcal{R}$-consistent and thus Δ is not conflict-free. □

Lemma 12. *Where $\Gamma \subseteq \mathcal{L}$, $\Delta^{\sim} = \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$, Δ is naive in $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$.*

Proof. Suppose $\Delta^{\sim} = \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$.

We first prove that Δ is conflict-free. Suppose for a contradiction, there is a $B \in Ab$ such that $\Gamma \cup \Delta \vdash_{\mathcal{R}} B, \overline{B}$. By Lemma 9, $\Gamma \cup \Delta \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} B, \overline{B}$. Hence, $\Gamma \cup \Delta \cup \mathcal{R}^{\sim}$ is $\mathbf{L}^3_{\mathcal{R}}$-inconsistent and by Fact 6 and compactness, $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} \bigvee \Theta^{\sim}$ for some finite $\Theta \subseteq \Delta$. Let Θ be $\subset$-minimal with this property, so that $\Theta^{\sim} \in \Sigma_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$. Since $\phi \cap \Theta^{\sim} = \emptyset$ this is a contradiction to the fact that $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$.

We now show that Δ is closed. Assume for a contradiction that $\Gamma \cup \Delta \vdash_{\mathcal{R}} A$ for some $A \in Ab \setminus \Delta$. By Lemma 9, $\Gamma \cup \Delta \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} A$. Note that ${\sim}A \in \phi$. By Fact 1, there is a $\Theta^{\sim} \in \Sigma_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$ for which $\{{\sim}A\} = \phi \cap \Theta^{\sim}$. Since $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} \bigvee \Theta^{\sim}$, by Fact 6, $\Gamma \cup \mathcal{R}^{\sim} \cup \Theta \setminus \{A\} \Vdash_{\mathbf{L}^3_{\mathcal{R}}} {\sim}A$. By the monotonicity of $\mathbf{L}^3_{\mathcal{R}}$, $\Gamma \cup \mathcal{R}^{\sim} \cup \Delta \Vdash_{\mathbf{L}^3_{\mathcal{R}}} {\sim}A$. Thus, $\Gamma \cup \Delta \cup \mathcal{R}^{\sim}$ is not $\mathbf{L}^3_{\mathcal{R}}$-consistent which implies by Lemma 10 that $\Gamma \cup \Delta$ is not $\mathcal{R}$-consistent. This contradicts the fact that Δ is conflict-free. □

The following theorem follows immediately in view of Lemma 11 and Lemma 12:

Theorem 10. *Where $\Gamma \subseteq \mathcal{L}$, Δ is a naive extension of $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$ iff $\Delta^{\sim} = \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$.*

If we suppose requirement (EX), we can also prove Theorem 9.

Lemma 13. *Where $\Gamma \subseteq \mathcal{L}$, $\mathcal{R}$ satisfies (EX), and $\Delta^{\sim} = \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$, Δ is stable in $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$.*

Proof. Suppose $\Delta^{\sim} = \Omega^{\sim}_{Ab} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}^3_{\mathcal{R}}}(\Gamma \cup \mathcal{R}^{\sim})$. In view of Lemma 12 we only need to show that Δ attacks all $B \in Ab \backslash \Delta$. Let thus $B \in Ab \backslash \Delta$. By (EX), $\Gamma \cup \Delta \vdash_{\mathcal{R}} \overline{B}$. Thus, Δ attacks B. □

The following Corollary follows immediately in view of Theorem 10 and Lemma 13.

Corollary 2. *Where $\mathcal{R}$ satisfies (EX), each naive set is stable in $\mathbf{ABA}^{Ab}_{\mathcal{R}}(\Gamma)$.*

In [7], the following was defined resp. proven:

Definition 27. *An assumption-based framework is normal iff every naive set of assumptions is stable.*

Theorem 11. *For any normal assumption-based framework, for any set of assumptions $\Delta \subseteq Ab$, Δ is naive iff Δ is stable iff Δ is preferred.*

[13] Or equivalently and analogous to the $\mathcal{R}$-consistency: if there is no $A \in \mathcal{L}_{\mathbf{L}}$ s.t. $\Gamma \nVdash_{\mathbf{L}^3_{\mathcal{R}}} A, \neg A$.

Corollary 3. *If an assumption-based framework satisfies (EX), $\Gamma \subseteq \mathcal{L}$, Δ is a preferred, stable and naive extension of $\mathbf{ABA}_{\mathcal{R}}^{Ab}(\Gamma)$ iff $\Delta^{\sim} = \Omega_{Ab}^{\sim} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}_{\mathcal{R}}^{3}}(\Gamma \cup \mathcal{R}^{\sim})$.*

We are now in a position to prove our two main theorems in this section.

Proof of Theorems 8 and 9. [Theorem 9.1, $\Leftarrow$]: Suppose that $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathsf{ns}}^{\Omega_{Ab}^{\sim}, \mathbf{L}_{\mathcal{R}}^{3}} A$. By Theorem 3, there is a $\Delta^{\sim} \subseteq \Omega_{Ab}^{\sim} \setminus \phi$ for some $\phi \in \Phi_{\mathbf{L}_{\mathcal{R}}^{3}}(\Gamma \cup \mathcal{R}^{\sim})$ s.t. $\Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}_{\mathcal{R}}^{3}} A \vee \bigvee \Delta^{\sim}$. By the monotonicity of $\mathbf{L}_{\mathcal{R}}^{3}$ and Fact 6, $\Theta \cup \Gamma \cup \mathcal{R}^{\sim} \Vdash_{\mathbf{L}_{\mathcal{R}}^{3}} A$ where $\Theta^{\sim} = \Omega_{Ab}^{\sim} \setminus \phi$. By Lemma 13, Θ is stable. Thus, $\Gamma \cup \Theta$ is $\mathcal{R}$-consistent. By Lemma 9, $\mathbf{ABA}_{\mathcal{R}}^{Ab}(\Gamma) \vdash_{\mathsf{sem}}^{\cup} A$.

The other direction and the other cases are shown analogously. □

9 Conclusion

In this paper we provided translations between several prominent systems in nonmonotonic logic (see Fig. 1 for an overview). In this conclusion we discuss some benefits.

In view of the translation of ALs into ABA we know that ALs can be understood as forms of formal argumentation. In view of the fact that ALs are equi-expressive with the syntactically characterised preferential semantics in Sec. 3 and Makinson's default assumptions, the same can be said about the latter two frameworks. Since a broad variety of defeasible reasoning forms in a wide range of application contexts have been explicated within the ALs family (see Sec. 2), all these reasoning forms are now available in the domain of formal argumentation. This may lead to further refinements. For instance, once embedded in ASPIC^{+} we gain rich resources to express preferences and priorities.

In view of the other direction, from a subclass of ABA to ALs, we know that this class can be understood in terms of the model-theoretic tools provided by KLM-style preferential semantics or, alternatively, as consistency management in terms of maximal consistent subsets as provided by default assumptions. This also means that meta-theoretic insights from, for instance, ALs become available for this subclass of ABA. For example, the computational complexity of ALs is well-understood [23, 15]. Moreover, properties of the AL consequence relations apply to this class of ABA. For instance, we know that adaptive consequence relations are cumulative (in the notation of Section 2, where AL is an adaptive logic, $\Gamma, \Delta, \{A\} \subseteq \mathcal{L}$, and $\Gamma \vdash_{\mathsf{AL}} B$ for all $B \in \Delta$, $\Gamma \vdash_{\mathsf{AL}} A$ iff $\Gamma \cup \Delta \vdash_{\mathsf{AL}} A$). For a study of meta-theoretic properties of ALs see [4, 19]. Finally, besides the available dialogue-based methods to model argumentative reasoning processes (e.g. [9]), now the dynamic proof theory of adaptive logics can also be used for this purpose.

Finally, we complete the circle between ABA and ASPIC^{+} (without priorities/preferences) by providing a translation from the latter to the former, whereas the other direction has been presented in [16]. As a side-product this provides a way to phrase the defeasible rules of ASPIC^{+} as strict rules. This shows that the strict fragment of ASPIC^{+} (without strict rules and thus without rebuttals and undercuts) is equi-expressive with full ASPIC^{+}. Such insights are conceptually interesting and may simplify future meta-theoretic investigations into ASPIC^{+}.

In future work we intend to generalise our investigations to approaches with priorities and preferences as provided in ASPIC^{+} and some generalisations of ALs. An interesting question will be, for instance, whether full ASPIC^{+} is translatable into lexicographic ALs [19, ch.5] or whether the latter can be translated to ABA or ASPIC^{+}.

References

[1] L. Amgoud and P. Besnard. Logical limits of abstract argumentation frameworks. *JANCL*, 23(3):229–267, 2013.

[2] O. Arieli and C. Straßer. Sequent-based logical argumentation. *A&C*, 6(1):73–99, 2015.

[3] D. Batens. Inconsistency-adaptive logics. *Logic at Work, Essays dedicated to the memory of Helena Rasiowa*, pages 445–472, 1999.

[4] D. Batens. A universal logic approach to adaptive logics. *Logica universalis*, 1(1):221–242, 2007.

[5] P. Besnard, A. Garcia, A. Hunter, S. Modgil, H. Prakken, G. Simari, and F. Toni. Introduction to structured argumentation. *A&C*, 5(1):1–4, 2014.

[6] P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *AI*, 128(1):203–235, 2001.

[7] A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *AI*, 93(1):63–101, 1997.

[8] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *AI*, 77:321–358, 1995.

[9] P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *AI*, 170(2):114–159, 2006.

[10] P. M. Dung, R. A. Kowalski, and F. Toni. Assumption-based argumentation. In *Argumentation in Artificial Intelligence*, pages 199–218. Springer, 2009.

[11] A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *TPLP*, 4(1+ 2):95–138, 2004.

[12] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *AI*, 44(1):167–207, 1990.

[13] D. Makinson. *Bridges from classical to nonmonotonic logic*. College Publications, 2005.

[14] S. Modgil and H. Prakken. The aspic+ framework for structured argumentation: a tutorial. *A&C*, 5(1):31–62, 2014.

[15] S. P. Odintsov and S. O. Speranski. Computability issues for adaptive logics in multi-consequence standard format. *SL*, 101(6):1237–1262, 2013.

[16] H. Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.

[17] N. Rescher and R. Manor. On inference from inconsistent premisses. *Theory Decis*, 1(2):179–217, 1970.

[18] Y. Shoham. Reasoning about change. Technical report, Yale Univ., New Haven, CT (USA), 1987.

[19] C. Straßer. *Adaptive Logics for Defeasible Reasoning*. Springer, 2014.

[20] F. Toni. A tutorial on assumption-based argumentation. *A&C*, 5(1):89–117, 2014.

[21] A. Urquhart. Basic many-valued logic. In *Handbook of philosophical logic*, pages 249–295. Springer, 2001.

[22] F. Van De Putte. Default assumptions and selection functions: a generic framework for non-monotonic logics. In *MICAI 2013*, pages 54–67. Springer, 2013.

[23] P. Verdée. Adaptive logics using the minimal abnormality strategy are π_1^1-complex. *Synthese*, 167(1):93–104, 2009.

Ordinal Conditional Functions for Nearly Counterfactual Revision

Aaron Hunter
BCIT
Burnaby, BC, Canada
aaron_hunter@bcit.ca

Abstract

We are interested in belief revision involving conditional statements where the antecedent is almost certainly false. To represent such problems, we use Ordinal Conditional Functions that may take infinite values. We model belief change in this context through simple arithmetical operations that allow us to capture the intuition that certain antecedents can not be validated by any number of observations. We frame our approach as a form of finite belief improvement, and we propose a model of conditional belief revision in which only the "right" hypothetical levels of implausibility are revised.

Introduction

The theory of belief change is concerned with the way agents incorporate new information. Typically, the focus is on new information that is given as a propositional formula. In this paper, we are concerned with situations where an agent needs to revise by a conditional where the antecedent is almost certainly false. More precisely, we consider antecedents that will not be believed given any finite amount of "regular" supporting evidence. We represent the degree of belief in such formulas using Ordinal Conditional Functions that may take infinite values, and we provide an approach to conditional revision based on basic ordinal arithmetic.

This paper makes several contributions to existing work on belief change.[1] First, we demonstrate that a simple algebra of belief change in the finite case extends naturally to the infinite case, giving a form of belief improvement. In the process, we demonstrate that there are natural examples in commonsense reasoning where multiple levels of infinite implausibility are actually useful. In particular, we introduce a natural approach to revision by conditional statements with little in the way of new formal machinery.

Motivating Example

Consider the following claims:

1. $heavy$: Your dog is overweight.
2. fly: Your dog can fly.
3. $hollow|fly$: If your dog can fly, then it has hollow bones.

The first two claims are simple declarative statements. But note that there is a clear difference in the amount of evidence needed to convince the agent to believe each claim. For (1), it presumably takes some finite number of reports from a trusted source. For (2), it seems unlikely that any finite number of reports would be convincing. This statement is almost certainly false, though it is possible to imagine a situation that would convince an agent to believe it.

The third statement is a conditional with a highly unlikely antecedent. Nevertheless, the perceived "impossibility" of (2) does not mean that (3) is free of content. Revision by (3) should change an agent's beliefs in a counterfactual sense; they may need to change their beliefs about hollow bones in some hypothetical scenario. Moreover, if ever the notion of flying dogs becomes believable, then this report will take on significance at the level of factual beliefs. In this paper, we refer to claims such as (3) as *nearly counterfactual*. We will provide a formal characterization of such claims, as well as a suitable approach to revision.

Preliminaries

Belief Revision

Belief revision is the belief change that occurs when new information is presented to an agent with some prior, possibly contradictory, set of beliefs. We assume an underlying propositional signature $\mathbf{P}$. An interpretation over $\mathbf{P}$ is called a *state*, while a logically closed set of formulas over $\mathbf{P}$ is called a *belief set*. A belief revision operator is a function that combines the initial belief set and a formula to produce a new belief set.

Formal approaches to belief revision typically require an agent to have some form of *ordering* or *ranking* that gives the relative plausibility of possible states. For example, in the well-known AGM approach, total pre-orders over states are used to represent the perceived likelihood of each state (AGM85; KM92). Unfortunately, this approach does not handle the problem of *iterated belief revision*. Related work has addressed iterated revision by explicitly specifying how the ordering changes, rather than just the belief set (DP97; BM06; JT07).

[1] This paper contains results that have been published in (Hun15) and (Hun16).

Ordinal Conditional Functions

An ordinal conditional function (OCF) is a function that maps each state to an ordinal (Spo88; Wil94). In this approach, strength of belief is captured by ordinal precedence. Hence, if r is an OCF and $r(s) < r(t)$, then s is a more plausible state than t. There is an obvious advantage to this approach in that a ranking function is clearly more expressive than a total pre-oder.

While the orginal definition allows the range of an OCF to be the class of all ordinals, in existing work it is common to restrict the range to the natural numbers, possibly with an additional symbol ∞ representing impossibility. In this paper, we will actually use a slightly larger range; so we need to briefly review ordinal arithmetic.[2] For our purposes, it is sufficient to note that ordinals are actually sets defined by an "order type." The finite ordinals are the natural numbers. The order type of the natural number n is unique, because it is the only ordinal that has exactly $n - 1$ preceding ordinals. The first infinite ordinal is ω, the set of all natural numbers. Every countably infinite subset of the natural numbers is order-isomorphic to ω.

It is easy to construct a countably infinite set that is not order isomorphic to ω: just add another symbol ∞ at the end that is larger than every natural number. The ordinal that defines the order type of this set is written $\omega + 1$. Similarly, there exists a distinct ordinal $\omega + n$ for any natural number n. And if we add a complete copy of the natural numbers, then we have the ordinal $\omega + \omega$ which is normally written as $\omega \cdot 2$. We can procede in this manner indefinitely to define a countably infinite sequence of ordinals. By taking powers, we can get even more order types; we will not delve further into this topic.

Ordinal addition can be understood in terms of the informal discussion above. Given ordinals α and β, the ordinal $\alpha + \beta$ has the order type obtained by taking a set with order type α and then appending a set with order type β where all the elements of β follow the elements of α in the underlying ordering. For finite ordinals, this coincides with the usual notion of addition. For infinite valued ordinals it does not. Note for example that $1 + \omega = \omega$; adding a number that precedes 0 does not change the order type, because the resulting structure is isomorphic to the natural numbers. On the other hand $\omega + 1 \neq \omega$. So ordinal addition is not commutative. It is also worth noting that ordinal subtraction is, in general, not well defined. In particular, it is not possible to define subtraction by ω.

Belief Change as Ordinal Arithmetic

Although our goal is to address revision by conditionals, we first introduce a simple approach to belief change based on the addition of ordinals. This will allow us to precisely define the notion of a nearly counterfactual statement, which is important for the class of conditionals that we wish to consider.

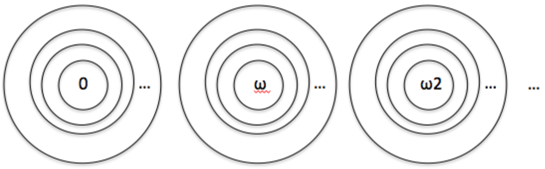


Figure 1: Visualizing ω^2

Restricted Domains

The following definition allows us to define conditional functions over any set of ordinals.

Definition 1 *Let S be a non-empty set of states and let Γ be a collection of ordinals. A Γ-CF (Γ conditional function) over S is a function $r : S \rightarrow \Gamma$ such that $r(s) = 0$ for some state s.*

Note that the definition of Γ-CFs does not actually specify that Γ is a *set*, because we do not wish to specify the underlying set theory in detail.

Several special cases are immediate:

- Spohn's ordinal conditional functions are Ω-CFs, where Ω is the collection of all ordinals.
- The class of ω-CFs coincides with the finite valued ranking functions common in the literature.
- The class of $(\omega + 1)$-CFs is the set of ranking functions that can take finite values, as well as the single "impossible" plausiblity value ∞. This is essentially equivalent to the possibilistic logic framework of (DP04), that uses the "necessity measure" of 0.

In this paper, we are primarily interested in the class of ω^2-CFs. Note that ω^2 can be specified as follows:

$$\omega^2 = \bigcup \{\omega \cdot k + c \mid k, c \in \omega\}.$$

Hence, every element of ω^2 can be written as $\omega \cdot k + c$ for some k and c. We think of these conditional functions as having countably many infinite levels of implausibility. A picture of ω^2 is shown in Figure 1.

If r is a Γ-CF, we write

$$Bel(r) = \{x \mid r(x) = 0\}.$$

The *degree of strength* of a conditional function r is the least n such that $n = r(v)$ for some $v \notin Bel(r)$. Hence, the degree of strength is a measure of how difficult it would be for an agent to abandon the currently believed set of states.

Finite Arithmetic on Conditional Functions

In the finite case, belief change can be captured through addition on ranking functions. Some variant of the following definition has appeared previously in published work by several authors; it is restated here and translated to our terminology.

[2] It is beyond the scope of this paper to give a complete treatment of infinite ordinals, and ordinal arithmetic. In the discussion here, we skip over fundamental set theory, and the fact that order-types are defined in terms of set-containment. We refer the reader to (Dev93) for an excellent introduction.

Definition 2 *Let r_1 and r_2 be ω-CFs over S, and let m be the minimum value of $r_1 + r_2$. Then $r_1 \bar{+} r_2$ is the function on S defined as follows:*

$$r_1 \bar{+} r_2(x) = r_1(x) + r_2(x) - m.$$

It is easy to check that this operation is associative, commutative, and that every element is invertible in the sense that, for each r there is an r' such that $r \bar{+} r' = 0$. Therefore, in terms of algebra, we say that the class of ω-CFs is an *abelian group* under $\bar{+}$.

Note that Spohn's *conditionalization* can be seen as a special case of this algebra on ranking functions. Let r_1 be a finite plausibility function representing the initial beliefs of an agent. Let ϕ be a formula, let d be a positive integer, and let r_2 be the ranking function defined as follows:

$$r_2(s) = \begin{cases} 0 \text{ if } s \models \phi \\ d \text{ otherwise} \end{cases}$$

Then $r_1 \bar{+} r_2$ is equivalent to Spohn's conditionalization of r_1 by ϕ with strength d. Similarly, if r_2 takes only two values and the degree of strength of r_2 is strictly larger than the degree of strength of r_1, then $r_1 \bar{+} r_2$ is AGM revision.

This approach does not extend to larger classes of ordinals.

Proposition 1 *Let β be an ordinal such that $\omega \in \beta$. Then $\bar{+}$ is not well-defined over the class of β-CFs.*

The problem is that subtraction is not defined for all pairs of (infinite) ordinals.

Example Consider the motivating example. We can define the following $(\omega + 1) - CFs$:

$$r_1(s) = \begin{cases} 0 \text{ if } s \models \{fly\} \\ \omega \text{ otherwise} \end{cases}$$

$$r_2(s) = \begin{cases} \omega \text{ if } s \models \{fly\} \\ 0 \text{ otherwise.} \end{cases}$$

Normalized addition of r_1 and r_2 requires us to calculate $\omega - \omega$. But this subtraction is not defined, so the calculation can not be completed.

This problem could be avoided by removing the normalization, but the result would no longer be an OCF. If we want to work with ranking functions that are closed under some form of addition, then we must either modify the definition, or we must relax the constraint that the pre-image of 0 is non-empty. We opt for the former.

Finite Zeroing

We define an algebra over ω^2-CFs based on *finite zeroing*. The following relation will be useful in proving results. In the definition, and in some future results, it is useful to consider functions over ordinals that do not necessarily take the value 0 for any argument. We use the general term Γ ranking to refer to an arbitrary function from S to Γ.[3]

Definition 3 *For Γ rankings r_1 and r_2, we write $r_1 \sim r_2$ just in case the following condition holds for every pair of states s, t*

$$r_1(s) < r_1(t) \iff r_2(s) < r_2(t).$$

Clearly, $\sim$ is an equivalence relation.

The intuition behind finite zeroing is that each conditional function can be categorized by its minimum value, in a manner that is useful for revision. Given any ω^2 ranking r, let $min(r)$ denote the minimum value $r(s)$. Note that a minimum is guaranteed by the fact that the ordinals are well-ordered.

Definition 4 *Let r be an ω^2 ranking with $\min(r) = \omega \cdot k + c$. Then k is the* degree *of r and c is the* finite shift, *written $deg(r)$ and $fin(r)$ respectively.*

We can use the degree and the finite shift to define the following operation.

Definition 5 *Let r be an ω^2 ranking with $deg(r) = k$ and $fin(r) = c$. Define $\bar{r}$ as follows. Let s be a state with $r(s) = \omega \cdot m + p$.*

1. *If $m > k$, then $\bar{r}(s) = \omega \cdot (m - k) + c$.*
2. *If $m = k$, then $\bar{r}(s) = (p - c)$.*

We call $\bar{r}$ the *finite zeroing* of r. Intuitively, elements at the "lowest level" are normalized to zero and elements at higher levels are shifted down by the degree of r. The following result is easy to prove.

Proposition 2 *If r is an ω^2 ranking, then $\bar{r}$ is a ω^2-CF and $r \sim \bar{r}$.*

Hence, the finite zeroing of any ranking is an equivalent ω^2-CF. We can now extend the definition of $*$ to ω^2-CFs.

Definition 6 *Let r_1, r_2 be ω^2-CFs. Then*

$$r_1 * r_2 = \overline{r_1 + r_2}.$$

Using this definition, $*$ is consistent with $\bar{+}$ for ω-CFs. Hence, $*$ can capture standard belief revision operators (e.g., AGM, DP) by restricting to finite values and setting the degree of strength of each function appropriately. This is the natural extension of revision, therefore, to the case that allows infinite plausibility values.

Example The motivating example over $\{heavy, fly\}$ can be captured by the following function:

$$r(s) = \begin{cases} \omega \text{ if } s \models fly \\ 10 \text{ if } s \models heavy \land \neg fly \\ 0 \text{ otherwise} \end{cases}$$

We let $*^n$ to denote a finite iteration of the $*$ operator. Suppose that, for each $V \in \{heavy, fly\}$, r_V is an OCF such that $r_V(s) = 2$ if and only if $s \not\models V$. The following are immediate:

- $r *^n r_{heavy}(s) = 0$ iff $n \geq 5$.
- $r *^n r_{fly}(s) \neq 0$ for any n.

[3] Konieczny refers to this kind of OCF as a *free OCF.*(Kon09)

Hence, it takes 5 reports to convince the owner that their dog is overweight. No finite number of reports will convince them that the dog can fly.

In the ω^2 case, the algebra obtained is not identical to the finite case.

Proposition 3 *The class of ω^2-CFs is a non-abelian group under $*$. (i.e. it is closed, associative, and every element has an inverse, but it is not commutative).*

The fact that $*$ is not commutative has interesting consequences, as illustrated in the following example.

Example Assume again that the vocabulary contains the predicates $\{heavy, fly\}$. Define

$$r_1(s) = \begin{cases} \omega \text{ if } s \models fly \\ 0 \text{ otherwise} \end{cases}$$

$$r_2(s) = \begin{cases} 0 \text{ if } s \models \neg heavy \wedge fly \\ 1 \text{ if } s \models heavy \wedge fly \\ 2 \text{ otherwise.} \end{cases}$$

Hence, r_1 says that an agent believes dogs can not fly; moreover the agent essentially believes that a flying dog is an impossibility. On the other hand, r_2 says that an agent believes that light dogs can fly - although the the strength of belief in this claim is only finite. Moreover, r_2 gives an ordering over less plausible states as well. Note that both r_1 and r_2 can be either an initial belief state or an observation. The following calculations are immediate.

$$r_1 * r_2(s) = \begin{cases} \omega \text{ if } s \models \neg heavy \wedge fly \\ \omega + 1 \text{ if } s \models heavy \wedge fly \\ 0 \text{ otherwise.} \end{cases}$$

$$r_2 * r_1(s) = \begin{cases} \omega \text{ if } s = \{fly\} \\ 0 \text{ otherwise.} \end{cases}$$

What is the significance of this example? It shows that conditional beliefs from an observation can be maintained at higher plausibility levels. In both cases, the underlying agent will not believe dogs can fly following revision. But the first revision allows the ordering of states to be refined somewhat at the conditional level. The second revision, on the other hand, washes away the finite level distinctions in the original belief set. This is similar to AGM revision in the sense that recent information seems to carry some particular weight. However, the infinite jumps in plausibility outweigh the preference for recency.

Nearly Counterfactual Reasoning

Motivation

In this section, we demonstrate how infinite-valued ordinal conditional functions can be useful for reasoning about conditional statements.

Example We return to the flying-dog example. Suppose that we initially believe $\neg fly$ and $\neg hollow$; in other words, we believe that dogs do not fly and that dogs do not have hollow bones. Now suppose we are told that flying dogs have hollow bones. Informaly, we want to revise by the conditional statement $(hollow|fly)$.

Note that $(hollow|fly)$ actually does not give any new information about dogs. This revision should not change the relative ordering of any worlds with a finite strength of belief. However, it does result in a change of belief. If one is later convinced of the existence of flying dogs, then the fact about hollow bones should be incorporated.

We refer to the reasoning in the preceding example as *nearly-counterfactual* revision. It is essentially a form of counterfactual reasoning, in which hypothetical worlds are considered in isolation. At the same time, however, we keep a form of conditional memory at higher ordinal levels. This is not only useful for perspective altering revelations, but we argue it can also be useful for analogical reasoning.

One important feature that is typically taken as a requirement for conditional reasoning is the Ramsey Test. In the context of revision by conditional statements, Kern-Isberner formulates the Ramsey Test as follows: when revising by a conditional, one would like to ensure that revision by $(\psi|\phi)$ followed by a revision by ϕ should guarantee belief in ψ (Ker99). We suggest that this formulation needs to be refined in order to be used in the case where infinite ranks are possible.

In the case of the flying dog, one is quite likely to accept the conditional $(hollow|fly)$ based on a single report with finite strength. However, a single report of fly with finite strength will not be believed. If the antecedent of the conditional is "very hard" to believe, then we should not expect the Ramsey Test to hold without some additional condition on the strength of the subsequent report. The problem, in a sense, is that the notion of believing a conditional is quite different than the notion of believing a fact. In order to believe $(hollow|fly)$, we simply need to keep some kind of record of this fact for the unlikely case where we discover that flying dogs happen to exist. On the other hand, in order to believe fly, we really need to make a significant change in our current world view.

Levels of Implausibility

Approaches to counterfactual reasoning are typically inspired to some degree by Lewis, who indicates that the truth of a counterfactual sentence is determined by its truth in alternative worlds (Lew73). We can represent this idea with ω^2-CFs. At each limit ordinal $\omega \cdot k$, we essentially have an entirely new plausibility ordering. As k increases, each such ordering represents an increasingly implausible world. However, a sufficiently strong observation can force our beliefs to jump to any of these unlikely worlds. As such, these are not truly counterfactual worlds, because we admit the possibility that they may eventually be believed.

The important property that we can capture with ω^2-CFs is the following: there are some formulas that may be true, yet we can not be convinced to believe them based on any finite number of pieces of "weak evidence." This allows us to give the following formal definition of the term *nearly counterfactual*.

Definition 7 *Let r be an OCF. A formula ϕ is nearly counterfactual with respect to r just in case there is no ω-CF r' such that $Bel(r * r') \models \phi$.*

The following is an immediate consequence of this definition.

Proposition 4 *If ϕ is nearly counterfactual with respect to r, then there is no finite sequence $r_1, \dots, r_n$ of ω-CFs such that $Bel(r * r_1 * \cdots * r_n) \models \phi$.*

We introduce some useful notation.

Definition 8 *Let ϕ be a formula. An OCF r is a ϕ-strengthening iff $Bel(r) = \{s \mid s \models \phi\}$.*

So, a ϕ-strengthening is just a ranking function where the minimal states are exactly the models of ϕ. For any formula ϕ, let (ϕ, n) be the ϕ-strengthening of ϕ where models of ϕ have plausibility 0 and every other state has plausibility n.

Definition 9 *Let r be an ω^2-CF. For any limit ordinal $\omega \cdot k$, let r_k be the following partial function:*

$$r_k(s) = \begin{cases} r(s), \textit{ if } r(s) = \omega \cdot k + c \textit{ for some } c \\ \textit{undefined otherwise} \end{cases}$$

Hence, r_k is just the restriction of r to those states with plausibility values at level k. We say that ϕ is *believed* at level k if $\{s \mid s \in \min(r_k)\} \models \phi$. Let $poss(\phi)$ denote the set of natural numbers k such that $s \models \phi$ for some s in the domain of r_k.

We can now introduce a form of strengthening with nearly counterfactual conditionals. In the definition, given an ω^2-CF r, we let $deg(s)$ denote the value k such that $r(s) = \omega \cdot k + c$.

Definition 10 *Let r be an ω^2-CF and let ψ, ϕ be formulas where ϕ is nearly counterfactual with respect to r. Let $n \in \omega$.*

$$r * (n, \psi|\phi)(s) = \begin{cases} r(s), \textit{ if } deg(s) \notin poss(\phi) \\ r * (\psi, n)(s) \textit{ otherwise} \end{cases}$$

We call this function the n-stengthening of ψ conditioned on ϕ. This function finds all levels of r where ϕ is possible, and then strengthens ψ at only those levels.

Example Let r again be the plausibility function

$$r(s) = \begin{cases} \omega \text{ if } s \models fly \\ 10 \text{ if } s \models heavy \wedge \neg fly \\ 0 \text{ otherwise} \end{cases}$$

It is easy to verify that fly is nearly counterfactual with respect to r. Now suppose that we extend the vocabulary to include the predicate symbol $hollow$. Define a new function r' as follows:

$$r'(s) = \begin{cases} r(s), \text{ if } s \not\models hollow \\ r(s) + 1, \text{ if } s \models hollow \end{cases}$$

This just says that we initally believe our dog does not have hollow bones; however, it is not particularly implausible. It follows that:

- $r'(s) = \omega$ if $s \models fly \wedge \neg hollow$.
- $r'(s) = \omega + 1$ if $s \models fly \wedge hollow$.

From these results, it follows that:

- $r' * (2, hollow|fly)(s) = r'(s)$, if $s \not\models fly$.
- $r' * (2, hollow|fly)(s) = \omega$, if $s \models fly \wedge hollow$.
- $r' * (2, hollow|fly)(s) = \omega + 1$, $s \models fly \wedge \neg hollow$.

So, roughly speaking, after strengthening by $(hollow|fly)$, we now believe that hollow bones are more plausible in all hypothetical situations where we believe flying dogs are possible.

Note that plausibility of a state is only changed at levels where ϕ is considered possible. Since the definition is only applied to nearly counterfactual conditions, this means that only hypothetical states are affected by the strengthening.

It remains to move from conditional strengthening to conditional revision. Recall that, for any ω^2-CF with $\min(r) = \omega \cdot k + c$, we write $fin(r) = c$.

Definition 11 *Let r be an ω^2-CF and let ψ, ϕ be formulas where ϕ is nearly counterfactual with respect to r.*

$$r * (\psi|\phi)(s) = \begin{cases} r(s), \textit{ if } deg(s) \notin poss(\phi) \\ r * (\psi, fin(r_k))(s) \textit{ if } r(s) = \omega \cdot k + c \end{cases}$$

Hence, for revision, we strengthen belief in ψ by the least value that will ensure ψ is believed at level k.

Under this definition, we satisfy a modified form of the Ramsey Test.

Proposition 5 *Let r be an ω^2-CF and let s be a state with $r(s) = \omega \cdot k + c$. If r' is an ω^2-CF with degree of strength larger than k and $Bel(r') \models \phi$, then*

$$Bel((r * (\psi|\phi)) * r') \models \psi.$$

Hence, if we revise by $(\psi|\phi)$ followed by an OCF with "sufficiently strong" belief in ϕ, then ψ will be believed.

Relation to Existing Work

Infinite Plausibility Values

There has been related work on the use of infinite valued ordinals in OCFs. In particular, Konieczny defines the notion of a *level* of belief explicitly in terms of limit ordinals(Kon09). In this work, different "levels" are used to represent beliefs that are independent in a precise sense. The lowest level is used for representing an agents actual beliefs about the world, whereas higher levels are used to represent integrity constraints. Our approach here is different in that we explicitly use the ordering on limit ordinals to represent infinite leaps in plausibility. This work is also distinguished by the fact that we use ordinal arithmetic on a small class of ordinals to define a simple algebra of belief change.

Belief Improvement

The success postulate ($K * \phi \vdash \phi$) of the AGM framework is clearly incorrect in cases where evidence is additive. That is to say, there are situations where a single observation is not sufficient to convince an agent to believe a particular fact.

Improvement operators (KP08) are belief change operators that address this issue by introducing a new set of postulates. The most important postulate states that an improvement operator $\circ$ must have the property that:

(I1) *There exists $n \in \mathbf{N}$ such that $B(\Psi \circ^n \phi) \vdash \phi$.*

Here Ψ is an epistemic state, and $B(\cdot)$ maps an epistemic state to the minimal elements of the underlying ordering. Hence, an improvement operator has the property that an agent will be convinced to believe ϕ after a finite number of improvements. The remaining postulates for a *weak improvement operator* are essentially the DP postulates applied to the operation $\circ^n$ obtained from **(I1)**. We refer the reader to (KP08) for the complete list of postulates.

We define an analog of **(I1)** as follows. If r_ϕ denotes a ϕ-strengthening, we can express the condition as follows.

(I*) *There exists $n \in \mathbf{N}$ such that $Bel(r *^n r_\phi) \models \phi$.*

The truth of this property depends on the degrees of strength of the functions.

Proposition 6 *If r is an ω-CF and r_ϕ is a ϕ-strengthening with finite strength, then $\mathbf{I}^*$ holds.*

For an epistemic state Ψ defined by $\prec_\Psi$, let r_Ψ be the *canonical representation* of Ψ.[4] Define $\circ_n$ such that $\Psi \circ \phi$ is obtained by taking the ordering induced by $r_\Psi * r(\phi, n)$.

Proposition 7 *For any $n \in \mathbf{N}$, the operator $\circ_n$ is a weak improvement operator.*

We call $\circ_n$ a *finite improvement operator*, because the degree of strength is finite. This result is essentially a corollary of Proposition 6, and it suggests that our $*$ operation based on normalized addition is actually the natural extension of *improvement* to the setting of ω-CFs.

The advantage of infinite plausibility values is that they give us greater flexibility in modelling improvement.

Proposition 8 *If r is an ω^2-CF and r_ϕ is a ϕ-strengthening with finite strength, then $\mathbf{I}^*$ does not hold.*

This result essentially states that **(I1)** is not a sound property for $*$ if we allow infinite plausibility values. This distinction can be seen in our running example. There is no finite number of improvements that will force the agent to believe that dogs can fly.

It is actually difficult to express the analog of Proposition 7 in the context of ω^2-CFs, because a total pre-order over states can not capture the "infinite jumps" in plausibility encoded by ω^2 ordinal ranks. But it is possible to define a correspondence between sequences of orderings and ordinals in ω^2.

Definition 12 *Let r be an ω^2-CF where $\max(r) = \omega \cdot d + b$ for some d, b. For $i \leq d$, let r_i denote the function defined as follows:*

1. *$domain(r_i) = \{s \mid r(s) = \omega \cdot i + c\}$.*
2. *If $r(s) = \omega \cdot i + c$, then $r_i(s) = c$.*

The following propositions are immediate.

Proposition 9 *Each r_i is a ω ranking, and there exists an ω-CF such that $r_i' \sim r_i$*

Proposition 10 *For any ω^2-CF r over a vocabulary $\mathbf{P}$ with $deg(r) = d$, there is an extended vocabulary $\mathbf{P}_1$ and a sequence $r_0, \ldots, r_d$ of ω-CFs such that, for each $i \leq d$, the r_i is equivalent (i.e. $\sim$) to the restriction of r to ordinals of degree i.*

This result is proved by just extending the vocabulary appropriately with propositional variables that make each infinite jump in the ordinal value definable. By breaking r into a set of ω-CFs, it follows that **(I*)** holds at level d when r_ϕ has degree of strength $\omega \cdot d$. Therefore, belief change by normalized addition on ω^2-CFs can really just be seen as a finite collection of improvements as each level. The important point, however, is that no finite sequence of improvements at level d will ever impact the actual beliefs at lower levels.

Conditional Belief Revision

Conditional belief revision was previously addressed by Kern-Isberner, who proposes a set of rationality postulates for conditional revision (Ker99). A concrete approach to conditional revision is also proposed, through the following ω-CF :

$$r * (\psi|\phi)(s) = \begin{cases} r(s) - r(\psi|\phi), \text{ if } s \models \phi \wedge \psi \\ r(s) + \alpha + 1, \text{ if } s \models \phi \wedge \neg\psi \\ r(s), \text{ if } s \models \neg\phi \end{cases}$$

where $\alpha = -1$ if $r(\{\phi, \psi\}) < r(\{\phi\})$, and $\alpha = 0$ otherwise. This operation satisfies all of the postulates for conditional revision, as well as the Ramsey Test. We remark, however, that this approach is not well-defined if we allow infinite plausibility values because of the ordinal subtraction on the right hand side. We suggest that this is not just a formal artefact of the theory; conditionals that are "almost certainly" false actually must be treated slightly differently.

In our approach, we essentially require the evidence for ϕ to be substantially stronger than the evidence for the conditional. We suggest that our beliefs following conditional revision should be changed in sort of an infinitesimally small way. While our beliefs about the actual world do not change, our beliefs about some (nearly) impossible world do, in fact, change.

Note that it is actually possible to reconcile our approach with Kern-Isberner's approach, by using the conditional revision above on each level r_k of the initial OCF r. At present, we are using a simple strengthening on each level, which actually flattens the plausibility structure after ordinal addition. A combined approach could respect the infinite jumps in plausibility, while satisfying the postulates for conditional revision at each level. We leave an investigation of this combined approach for future work.

Discussion

Conclusion

In this paper, we have explored the use of infinite ordinals for reasoning about belief change and conditional reasoning. We have shown that allowing plausibility values to range

[4] If s is in the n^{th} level of Ψ , then $r_\Psi(s) = n$

over ω^2 results in a belief algebra that is only slightly more complicated, and we gain an expressive advantage. In particular, we can represent situations where stubbornly held beliefs are resistant to evidence to the contrary. We have demonstrated that this results in a slightly more expressive class of improvement operators where evidence increases relative belief, but no finite number of improvements will actually lead to a change in the belief state. Finally, we addressed so-called "nearly counterfactual" revision, where we incorporate information that is conditional on a highly unlikely statement.

Future Work

This paper is a preliminary exploration into different applications and formal properties of infinite valued ordinal conditional functions. It remains to move beyond ω^2-CFs, to completely characterize the relationship with improvement operators, and to consider further practical applications.

In the present framework, we have discussed nearly counterfactual reasoning as a tool for keeping a sort of "memory" about unlikely situations, in order to incorporate this information later if necessary. But there is also a natural kind of reasoning that would allow us to use conditionals to reason by analogy about the actual state of the world. Consider the following well-known ambiguity from (Lew73), and originally attributed to Quine:

1. If Caesar was president, he would use nuclear weapons.
2. If Caesar was president, he would use catapults.

As a conditional, we could write both as $(W|C)$, where W stands for a weapon that would be used and C is the condition "Caesar is president." But (1) suggests that we condition by imagining Caesar alive in the current world. So this is a conditional statement interpreted in the current state of the world. On the other hand, (2) suggests that we consider what would happen in some past world where Caesar exists.

Now suppose that we believe a certain politician is actually very similar to Caesar. If we believe that Caesar would use nuclear weapons, then we may conclude that this "real" politician would also use nuclear weapons. Formally, we could proceed as follows: if some hypothetical world is isomorphic to the current state of the world when we restrict the vocabulary (to not include Caesar), then we can use inferences about the hypothetical world to draw conclusions about the actual world. This is a form of *ampliative reasoning* that we intend to explore through ω^2-CFs in future work.

References

C.E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet functions for contraction and revision. *Journal of Symbolic Logic*, 50(2):510–530, 1985.

R. Booth and T. Meyer. Admissible and restrained revision. *Journal of Artificial Intelligence Research*, 26:127–151, 2006.

K. Devlin. *The Joy of Sets*. Springer, 1993.

A. Darwiche and J. Pearl. On the logic of iterated belief revision. *Artificial Intelligence*, 89(1-2):1–29, 1997.

Didier Dubois and Henri Prade. Possibilistic logic: a retrospective and prospective view. *Fuzzy Sets and Systems*, 144(1):3–23, 2004.

A. Hunter. Infinite ordinals and finite improvement. In *Proceedings of the International Conference on Logic, Interaction and Rationality (LORI)*, pages 416–420, 2015.

A. Hunter. Nearly counterfactual revision. In *Proceedings of the Canadian Conference on Artificial Intelligence*, 2016.

Y. Jin and M. Thielscher. Iterated belief revision, revised. *Artificial Intelligence*, 171(1):1–18, 2007.

Gabriele Kern-Isberner. Postulates for conditional belief revision. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 186–191, 1999.

H. Katsuno and A.O. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(2):263–294, 1992.

S. Konieczny. Using transfinite ordinal conditional functions. In *Proceedings of Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 10th European Conference, ECSQARU 2009*, pages 396–407, 2009.

Sébastien Konieczny and Ramón Pino Peréz. Improvement operators. In *Eleventh International Conference on Principles of Knowledge Representation and Reasoning (KR'08)*, pages 177–186, 2008.

D. Lewis. *Counterfactuals*. Harvard University Press, 1973.

W. Spohn. Ordinal conditional functions. A dynamic theory of epistemic states. In W.L. Harper and B. Skyrms, editors, *Causation in Decision, Belief Change, and Statistics, vol. II*, pages 105–134. Kluwer Academic Publishers, 1988.

M. A. Williams. Transmutations of knowledge systems. In *Proceedings of the Fourth International Conference on the Principles of Knowledge Representation and Reasoning (KR94)*, pages 619–629, 1994.

Characterizing Realizability in Abstract Argumentation[*]

Thomas Linsbichler
TU Wien
Austria

Jörg Pührer and **Hannes Strass**
Leipzig University
Germany

Abstract

Realizability for knowledge representation formalisms studies the following question: Given a semantics and a set of interpretations, is there a knowledge base whose semantics coincides exactly with the given interpretation set? We introduce a general framework for analyzing realizability in abstract dialectical frameworks (ADFs) and various of its subclasses. In particular, the framework applies to Dung argumentation frameworks, SETAFs by Nielsen and Parsons, and bipolar ADFs. We present a uniform characterization method for the admissible, complete, preferred and model/stable semantics. We employ this method to devise an algorithm that decides realizability for the mentioned formalisms and semantics; moreover the algorithm allows for constructing a desired knowledge base whenever one exists. The algorithm is built in a modular way and thus easily extensible to new formalisms and semantics. We have also implemented our approach in answer set programming, and used the implementation to obtain several novel results on the relative expressiveness of the abovementioned formalisms.

1 Introduction

The abstract argumentation frameworks (AFs) introduced by Dung (1995) have garnered increasing attention in the recent past. In his seminal paper, Dung showed how an abstract notion of argument (seen as an atomic entity) and the notion of individual attacks between arguments together could reconstruct several established KR formalisms in argumentative terms. Despite the generality of those and many more results in the field that was sparked by that paper, researchers also noticed that the restriction to *individual attacks* is often overly limiting, and devised extensions and generalizations of Dung's frameworks: directions included generalizing individual attacks to *collective attacks* (Nielsen and Parsons, 2006), leading to so-called SETAFs; others started offering a *support* relation between arguments (Cayrol and Lagasquie-Schiex, 2005), preferences among arguments (Amgoud and Cayrol, 2002; Modgil, 2009), or attacks on attacks into arbitrary depth (Baroni et al., 2011). This is only the tip of an iceberg, for a more comprehensive overview we refer to the work of Brewka, Polberg, and Woltran (2014).

One of the most recent and most comprehensive generalizations of AFs has been presented by Brewka and Woltran (2010) (and later continued by Brewka et al., 2013) in the form of *abstract dialectical frameworks (ADFs)*. These ADFs offer any type of link between arguments: individual attacks (as in AFs), collective attacks (as in SETAFs), and individual and collective support, to name only a few. This generality is achieved through so-called *acceptance conditions* associated to each statement. Roughly, the meaning of relationships between arguments is not fixed in ADFs, but is specified by the user for each argument in the form of Boolean functions (acceptance functions) on the argument's parents. However, this generality comes with a price: Strass and Wallner (2015) found that the complexity of the associated reasoning problems of ADFs is in general higher than in AFs (one level up in the polynomial hierarchy). Fortunately, the subclass of *bipolar ADFs* (defined by Brewka and Woltran, 2010) is as complex as AFs (for all considered semantics) while still offering a wide range of modeling capacities (Strass and Wallner, 2015). However, there has only been little concerted effort so far to exactly analyze and compare the expressiveness of the abovementioned languages.

This paper is about exactly analyzing means of expression for argumentation formalisms. Instead of motivating expressiveness in natural language and showing examples that some formalisms seem to be able to express but others do not, we tackle the problem in a formal way. We use a precise mathematical definition of expressiveness: a set of interpretations is *realizable* by a formalism under a semantics if and only if there exists a knowledge base of the formalism whose semantics is exactly the given set of interpretations. Studying realizability in AFs has been started by Dunne et al. (2013, 2015), who analyzed realizability for extension-based semantics, that is, interpretations represented by sets where arguments are either accepted (in the extension set) or not accepted (not in the extension set). While their initial work disregarded arguments that are never accepted, there have been continuations where the existence of such "invisible" arguments is ruled out (Baumann et al., 2014; Linsbichler, Spanring, and Woltran, 2015). Dyrkolbotn (2014) began to analyze realizability for labeling-based semantics of AFs, that is, three-valued semantics where arguments can be accepted (mapped to true), rejected (mapped to false) or neither (mapped to unknown). Strass (2015) started to ana-

[*] This research has been supported by DFG (project BR 1817/7-1) and FWF (projects I1102 and P25518).

lyze the relative expressiveness of two-valued semantics for ADFs (relative with respect to related formalisms). Most recently, Pührer (2015) presented precise characterizations of realizability for ADFs under several three-valued semantics, namely admissible, grounded, complete, and preferred. The term "precise characterizations" means that he gave necessary and sufficient conditions for an interpretation set to be ADF-realizable under a semantics.

The present paper continues this line of work by lifting it to a much more general setting. We combine the works of Dunne et al. (2015), Pührer (2015), and Strass (2015) into a unifying framework, and at the same time extend them to formalisms and semantics not considered in the respective papers: we treat several formalisms, namely AFs, SETAFs, and (B)ADFs, while the previous works all used different approaches and techniques. This is possible because all of these formalisms can be seen as subclasses of ADFs that are obtained by suitably restricting the acceptance conditions.

Another important feature of our framework is that we uniformly use three-valued interpretations as the underlying model theory. In particular, this means that arguments cannot be "invisible" any more since the underlying vocabulary of arguments is always implicit in each interpretation. Technically, we always assume a fixed underlying vocabulary and consider our results parametric in that vocabulary. In contrast, for example, Dyrkolbotn (2014) presents a construction for realizability that introduces new arguments into the realizing knowledge base; we do not allow that. While sometimes the introduction of new arguments can make sense, for example if new information becomes available about a domain or a debate, it is not sensible in general, as these new arguments would be purely technical with an unclear dialectical meaning. Moreover, it would lead to a different notion of realizability, where most of the realizability problems would be significantly easier, if not trivial.

The paper proceeds as follows. We begin with recalling and introducing the basis and basics of our work – the formalisms we analyze and the methodology with which we analyze them. Next we introduce our general framework for realizability; the major novelty is our consistent use of so-called characterization functions, firstly introduced by Pührer (2015), which we adapt to further semantics. The main workhorse of our approach will be a parametric propagate-and-guess algorithm for deciding whether a given interpretation set is realizable in a formalism under a semantics. We then analyze the relative expressiveness of the considered formalisms, presenting several new results that we obtained using an implementation of our framework. We conclude with a discussion.

2 Preliminaries

We make use of standard mathematical concepts like functions and partially ordered sets. For a function $f : X \to Y$ we denote the *update of f with a pair* $(x, y) \in X \times Y$ by $f|_y^x : X \to Y$ with $z \mapsto y$ if $z = x$, and $z \mapsto f(z)$ otherwise. For a function $f : X \to Y$ and $y \in Y$, its preimage is $f^{-1}(y) = \{x \in X \mid f(x) = y\}$. A *partially ordered set* is a pair $(S, \sqsubseteq)$ with $\sqsubseteq$ a partial order on S. A partially ordered set $(S, \sqsubseteq)$ is a *complete lattice* if and only if every $S' \subseteq S$ has both a greatest lower bound (glb) $\bigsqcap S' \in S$ and a least upper bound (lub) $\bigsqcup S' \in S$. A partially ordered set $(S, \sqsubseteq)$ is a *complete meet-semilattice* iff every non-empty subset $S' \subseteq S$ has a greatest lower bound $\bigsqcap S' \in S$ (the *meet*) and every ascending chain $C \subseteq S$ has a least upper bound $\bigsqcup C \in S$.

Three-Valued Interpretations Let A be a fixed finite set of statements. An *interpretation* is a mapping $v : A \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$ that assigns one of the truth values true $(\mathbf{t})$, false $(\mathbf{f})$ or unknown $(\mathbf{u})$ to each statement. An interpretation is *two-valued* if $v(A) \subseteq \{\mathbf{t}, \mathbf{f}\}$, that is, the truth value $\mathbf{u}$ is not assigned. Two-valued interpretations v can be extended to assign truth values $v(\varphi) \in \{\mathbf{t}, \mathbf{f}\}$ to propositional formulas φ as usual.

The three truth values are partially ordered according to their information content: we have $\mathbf{u} <_i \mathbf{t}$ and $\mathbf{u} <_i \mathbf{f}$ and no other pair in $<_i$, which intuitively means that the classical truth values contain more information than the truth value unknown. As usual, we denote by $\leq_i$ the partial order associated to the strict partial order $<_i$. The pair $(\{\mathbf{t}, \mathbf{f}, \mathbf{u}\}, \leq_i)$ forms a complete meet-semilattice with the information meet operation $\sqcap_i$. This meet can intuitively be interpreted as *consensus* and assigns $\mathbf{t} \sqcap_i \mathbf{t} = \mathbf{t}$, $\mathbf{f} \sqcap_i \mathbf{f} = \mathbf{f}$, and returns $\mathbf{u}$ otherwise.

The information ordering $\leq_i$ extends in a straightforward way to interpretations v_1, v_2 over A in that $v_1 \leq_i v_2$ iff $v_1(a) \leq_i v_2(a)$ for all $a \in A$. We say for two interpretations v_1, v_2 that v_2 *extends* v_1 iff $v_1 \leq_i v_2$. The set $\mathcal{V}$ of all interpretations over A forms a complete meet-semilattice with respect to the information ordering $\leq_i$. The consensus meet operation $\sqcap_i$ of this semilattice is given by $(v_1 \sqcap_i v_2)(a) = v_1(a) \sqcap_i v_2(a)$ for all $a \in A$. The least element of $(\mathcal{V}, \leq_i)$ is the valuation $v_{\mathbf{u}} : A \to \{\mathbf{u}\}$ mapping all statements to unknown – the least informative interpretation. By $\mathcal{V}_2$ we denote the set of two-valued interpretations; they are the $\leq_i$-maximal elements of the meet-semilattice $(\mathcal{V}, \leq_i)$. We denote by $[v]_2$ the set of all two-valued interpretations that extend v. The elements of $[v]_2$ form an $\leq_i$-antichain with greatest lower bound $v = \bigsqcap_i [v]_2$.

Abstract Argumentation Formalisms An *abstract dialectical framework (ADF)* is a tuple $D = (A, L, C)$ where A is a set of statements (representing positions one can take or not take in a debate), $L \subseteq A \times A$ is a set of links (representing dependencies between the positions), $C = \{C_a\}_{a \in A}$ is a collection of functions $C_a : 2^{par(a)} \to \{\mathbf{t}, \mathbf{f}\}$, one for each statement $a \in A$. The function C_a is the *acceptance condition of a* and expresses whether a can be accepted, given the acceptance status of its parents $par(a) = \{b \in S \mid (b, a) \in L\}$. We usually represent each C_a by a propositional formula φ_a over $par(a)$. To specify an acceptance condition, then, we take $C_a(M \cap par(a)) = \mathbf{t}$ to hold iff M is a model for φ_a.

Brewka and Woltran (2010) introduced a useful subclass of ADFs: an ADF $D = (A, L, C)$ is *bipolar* iff all links in L are supporting or attacking (or both). A link $(b, a) \in L$ is *supporting in D* iff for all $M \subseteq par(a)$, we have that $C_a(M) = \mathbf{t}$ implies $C_a(M \cup \{b\}) = \mathbf{t}$. Symmetrically, a link $(b, a) \in L$ is *attacking in D* iff for all $M \subseteq par(a)$, we have that $C_a(M \cup \{b\}) = \mathbf{t}$ implies $C_a(M) = \mathbf{t}$. If a link (b, a) is both supporting and attacking then b has no actual influence on a. (But the link does not violate bipolarity.) We write BADFs as $D = (A, L^+ \cup L^-, C)$ and mean that L^+ contains all supporting links and L^- all attacking links.

The semantics of ADFs can be defined using an operator Γ_D over three-valued interpretations (Brewka and Woltran, 2010; Brewka et al., 2013). For an ADF D and a three-valued interpretation v, the interpretation $\Gamma_D(v)$ is given by

$$a \mapsto \textstyle\bigsqcap_i \{w(\varphi_a) \mid w \in [v]_2\}$$

That is, for each statement a, the operator returns the consensus truth value for its acceptance formula φ_a, where the consensus takes into account all possible two-valued interpretations w that extend the input valuation v. If this v is two-valued, we get $[v]_2 = \{v\}$ and thus $\Gamma_D(v)(a) = v(\varphi_a)$.

The standard semantics of ADFs are now defined as follows. For ADF D, an interpretation $v : A \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$ is

- *admissible* iff $v \leq_i \Gamma_D(v)$;
- *complete* iff $\Gamma_D(v) = v$;
- *preferred* iff it is $\leq_i$-maximal admissible;
- a *two-valued model* iff it is two-valued and $\Gamma_D(v) = v$.

We denote the sets of interpretations that are admissible, complete, preferred, and two-valued models by $adm(D)$, $com(D)$, $prf(D)$ and $mod(D)$, respectively. These definitions are proper generalizations of Dung's notions for AFs: For an AF (A, R), where $R \subseteq A \times A$ is the attack relation, the *ADF associated to* (A, R) is $D_{(A,R)} = (A, R, C)$ with $C = \{\varphi_a\}_{a \in A}$ and $\varphi_a = \bigwedge_{b:(b,a) \in R} \neg b$ for all $a \in A$. AFs inherit their semantics from the definitions for ADFs (Brewka et al., 2013, Theorems 2 and 4). In particular, an interpretation is *stable* for an AF (A, R) if and only if it is a two-valued model of $D_{(A,R)}$.

A SETAF is a pair $S = (A, X)$ where $X \subseteq (2^A \setminus \{\emptyset\}) \times A$ is the (set) attack relation. We define three-valued counterparts of the semantics introduced by Nielsen and Parsons (2006), following the same conventions as in three-valued semantics of AFs (Caminada and Gabbay, 2009) and argumentation formalisms in general. Given a statement $a \in A$ and an interpretation v we say that a is *acceptable* wrt. v if $\forall (B, a) \in X \exists a' \in B : v(a') = \mathbf{f}$ and a is *unacceptable* wrt. v if $\exists (B, a) \in X \forall a' \in B : v(a') = \mathbf{t}$. For an interpretation $v : A \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$ it holds that

- $v \in adm(S)$ iff for all $a \in A$, a is acceptable wrt. v if $v(a) = \mathbf{t}$ and a is unacceptable wrt. v if $v(a) = \mathbf{f}$;
- $v \in com(S)$ iff for all $a \in A$, a is acceptable wrt. v iff $v(a) = \mathbf{t}$ and a is unacceptable wrt. v iff $v(a) = \mathbf{f}$;
- $v \in prf(S)$ iff v is $\leq_i$-maximal admissible; and
- $v \in mod(S)$ iff $v \in adm(F)$ and $\nexists a \in A : v(a) = \mathbf{u}$.

For a SETAF $S = (A, X)$ the corresponding ADF D_S has acceptance formula $\varphi_a = \bigwedge_{(B,a) \in X} \bigvee_{a' \in B} \neg a'$ for each statement $a \in A$. (Polberg, 2016)

Proposition 1. *For any SETAF $S = (A, X)$ it holds that $\sigma(S) = \sigma(D_S)$, where $\sigma \in \{adm, com, prf, mod\}$.*

Proof. Given interpretation v and statement a, it holds that $\Gamma_{D_S}(v)(a) = \mathbf{t}$ iff $\forall w \in [v]_2 : w(a) = \mathbf{t}$ iff $\forall (B, a) \in X$ $\exists a' \in B : v(a') = \mathbf{f}$ iff a is acceptable wrt. v and $\Gamma_{D_S}(v)(a) = \mathbf{f}$ iff $\forall w \in [v]_2 : w(a) = \mathbf{f}$ iff $\exists (B, a) \in X$ $\forall a' \in B : v(a') = \mathbf{t}$ iff a is unacceptable wrt. v. Hence $\sigma(S) = \sigma(D_S)$ for $\sigma \in \{adm, com, prf, mod\}$. □

Realizability A set $V \subseteq \mathcal{V}$ of interpretations is *realizable* in a formalism $\mathcal{F}$ under a semantics σ if and only if there exists a knowledge base $\mathsf{kb} \in \mathcal{F}$ having exactly $\sigma(\mathsf{kb}) = V$. Pührer (2015) characterized realizability for ADFs under various three-valued semantics. We will reuse the central notions for capturing the complete semantics in this work.

Definition 1 (Pührer 2015). Let V be a set of interpretations. A function $f : \mathcal{V}_2 \to \mathcal{V}_2$ is a *com-characterization* of V iff: for each $v \in \mathcal{V}$ we have $v \in V$ iff for each $a \in A$:

- $v(a) \neq \mathbf{u}$ implies $f(v_2)(a) = v(a)$ for all $v_2 \in [v]_2$ and
- $v(a) = \mathbf{u}$ implies $f(v_2')(a) = \mathbf{t}$ and $f(v_2'')(a) = \mathbf{f}$ for some $v_2', v_2'' \in [v]_2$. ▲

From a function of this kind we can build a corresponding ADF by the following construction. For a function $f : \mathcal{V}_2 \to \mathcal{V}_2$, we define D_f as the ADF where the acceptance formula for each statement a is given by

$$\varphi_a^f = \bigvee_{\substack{w \in \mathcal{V}_2,\\ f(w)(a) = \mathbf{t}}} \phi_w \quad \text{with} \quad \phi_w = \bigwedge_{w(a') = \mathbf{t}} a' \wedge \bigwedge_{w(a') = \mathbf{f}} \neg a'$$

Observe that we have $v(\phi_w) = \mathbf{t}$ iff $v = w$ by definition. Intuitively, the acceptance condition φ_a^f is constructed such that v is a model of φ_a^f if and only if we find $f(v)(a) = \mathbf{t}$.

Proposition 2 (Pührer 2015). *Let $V \subseteq \mathcal{V}$ be a set of interpretations. (1) For each ADF D with $com(D) = V$, there is a com-characterization f_D for V; (2) for each com-characterization $f : \mathcal{V}_2 \to \mathcal{V}_2$ for V we have $com(D_f) = V$.*

The result shows that V can be realized under complete semantics if and only if there is a com-characterization for V.

3 A General Framework for Realizability

The main underlying idea of our framework is that all abstract argumentation formalisms introduced in the previous section can be viewed as subclasses of abstract dialectical frameworks. This is clear for ADFs themselves and for BADFs by definition; for AFs and SETAFs it is fairly easy to see. However, knowing that these formalisms can be recast as ADFs is not everything. To employ this knowledge for realizability, we must be able to precisely characterize the corresponding subclasses in terms of restricting the ADFs' acceptance functions. Alas, this is also possible and paves the way for the framework we present in this section. Most importantly, we will make use of the fact that different formalisms and different semantics can be characterized modularly, that is, independently of each other.

Towards a uniform account of realizability for ADFs under different semantics, we start with a new characterization of realizability for ADFs under admissible semantics that is based on a notion similar in spirit to com-characterizations.

Definition 2. Let V be a set of interpretations. A function $f : \mathcal{V}_2 \to \mathcal{V}_2$ is an *adm-characterization* of V iff: for each $v \in \mathcal{V}$ we have $v \in V$ iff for every $a \in A$:

- $v(a) \neq \mathbf{u}$ implies $f(v_2)(a) = v(a)$ for all $v_2 \in [v]_2$. ▲

Note that the only difference to Definition 1 is dropping the second condition related to statements with truth value $\mathbf{u}$.

Proposition 3. *Let $V \subseteq \mathcal{V}$ be a set of interpretations. (1) For each ADF D such that $adm(D) = V$, there is an adm-characterization f_D for V; (2) for each adm-characterization $f : \mathcal{V}_2 \to \mathcal{V}_2$ for V we have $adm(D_f) = V$.*

Proof. (1) We define the function $f_D : \mathcal{V}_2 \to \mathcal{V}_2$ as $f_D(v_2)(a) = v_2(\varphi_a)$ for every $v_2 \in \mathcal{V}_2$ and $a \in A$ where φ_a is the acceptance formula of a in D. We will show that f_D is an adm-characterization for $V = adm(D)$. Let v be an interpretation. Consider the case $v \in adm(D)$ and $v(a) \neq u$ for some $a \in A$ and some $v_2 \in [v]_2$. From $v \leq_i \Gamma_D(v)$ we get $v_2(\varphi_a) = v(a)$. By definition of f_D is follows that $f_D(v_2)(a) = v(a)$. Now assume $v \notin adm(D)$ and consequently $v \not\leq_i \Gamma_D(v)$. There must be some $a \in A$ such that $v(a) \neq \mathbf{u}$ and $v(a) \neq \Gamma_D(v)(a)$. Hence, there is some $v_2 \in [v]_2$ with $v_2(\varphi_a) \neq v(a)$ and $f_D(v_2)(a) \neq v(a)$ by definition of f_D. Thus, f_D is an adm-characterization

(2) Observe that for every two-valued interpretation v_2 and every $a \in A$ we have $f(v_2)(a) = v_2(\varphi_a^f)$. ($\subseteq$): Let $v \in adm(D_f)$ be an interpretation and $a \in A$ a statement such that $v(a) \neq \mathbf{u}$. Let v_2 be a two-valued interpretation with $v_2 \in [v]_2$. Since $v \leq_i \Gamma_{D_f}(v)$ we have $v(a) = v_2(\varphi_a^f)$. Therefore, by our observation it must also hold that $f(v_2)(a) = v(a)$. Thus, by Definition 2, $v \in V$. ($\supseteq$): Consider an interpretation v such that $v \notin adm(D_f)$. We show that $v \notin V$. From $v \notin adm(D_f)$ we get $v \not\leq_i \Gamma_{D_f}(v)$. There must be some $a \in A$ such that $v(a) \neq \mathbf{u}$ and $v(a) \neq \Gamma_{D_f}(v)(a)$. Hence, there is some $v_2 \in [v]_2$ with $v_2(\varphi_a^f) \neq v(a)$ and consequently $f(v_2)(a) \neq v(a)$. Thus, by Definition 2 we have $v \notin V$. □

When listing sets of interpretations in examples, for the sake of readability we represent three-valued interpretations by sequences of truth values, tacitly assuming that the underlying vocabulary is given and has an associated total ordering. For example, for the vocabulary $A = \{a, b, c\}$ we represent the interpretation $\{a \mapsto \mathbf{t}, b \mapsto \mathbf{f}, c \mapsto \mathbf{u}\}$ by the sequence $\mathbf{tfu}$.

Example 1. Consider the sets $V_1 = \{\mathbf{uuu}, \mathbf{tff}, \mathbf{ftu}\}$ and $V_2 = \{\mathbf{tff}, \mathbf{ftu}\}$ of interpretations over $A = \{a, b, c\}$. The mapping $f = \{\mathbf{ttt} \mapsto \mathbf{ftt}, \mathbf{ttf} \mapsto \mathbf{tft}, \mathbf{tft} \mapsto \mathbf{ttt}, \mathbf{tff} \mapsto \mathbf{tff}, \mathbf{ftt} \mapsto \mathbf{ftf}, \mathbf{ftf} \mapsto \mathbf{ftt}, \mathbf{fft} \mapsto \mathbf{ttf}, \mathbf{fff} \mapsto \mathbf{ftf}\}$ is an adm-characterization for V_1. Thus, the ADF D_f has V_1 as its admissible interpretations. Indeed, the realizing ADF has the following acceptance conditions:

$$\begin{aligned}
\varphi_a^f &\equiv (a \wedge b \wedge \neg c) \vee (a \wedge \neg b) \vee (\neg a \wedge \neg b \wedge c)\\
\varphi_b^f &\equiv (a \wedge c) \vee (\neg a \wedge b) \vee (\neg a \wedge \neg b \wedge \neg c)\\
\varphi_c^f &\equiv (a \wedge b) \vee (\neg a \wedge b \wedge \neg c) \vee (\neg b \wedge c)
\end{aligned}$$

For V_2 no adm-characterization exists because $\mathbf{uuu} \notin V_2$ but the implication of Definition 2 trivially holds for a, b, and c. ■

We have seen that the construction D_f for realizing under complete semantics can also be used for realizing a set V of interpretations under admissible semantics. The only difference is that we here require f to be an adm-characterization instead of a com-characterization for V. Note that admissible semantics can be characterized by properties that are easier to check than existence of an adm-characterization (see the work of Pührer, 2015). However, using the same type of characterizations for different semantics allows for a unified approach for checking realizability and constructing a realizing ADF in case one exists.

For realizing under the model semantics, we can likewise present an adjusted version of com-characterizations.

Definition 3. Let $V \subseteq \mathcal{V}$ be a set of interpretations. A function $f : \mathcal{V}_2 \to \mathcal{V}_2$ is a *mod-characterization* of V if and only if: (1) f is defined on V (that is, $V \subseteq \mathcal{V}_2$) and (2) for each $v \in \mathcal{V}_2$, we have $v \in V$ iff $f(v) = v$. ▲

As we can show, there is a one-to-one correspondence between mod-characterizations and ADF realizations.

Proposition 4. *Let $V \subseteq \mathcal{V}$ be a set of interpretations. (1) For each ADF D such that $mod(D) = V$, there is a mod-characterization f_D for V; (2) vice versa, for each mod-characterization $f : \mathcal{V}_2 \to \mathcal{V}_2$ for V we find $mod(D_f) = V$.*

Proof. (1) Let D be an ADF with $mod(D) = V$. It immediately follows that $V \subseteq \mathcal{V}_2$. To define f_D we can use the construction in the proof of Proposition 3. It follows directly that for any $v \in \mathcal{V}_2$, we find $f_D(v) = v$ iff $v \in V$. Thus f_D is a mod-characterization for V.

(2) Let $V \subseteq \mathcal{V}_2$ and $f : \mathcal{V}_2 \to \mathcal{V}_2$ be a mod-characterization of V. For any $v \in \mathcal{V}_2$ we have:

$$\begin{aligned}
& v \in V \iff v = f(v)\\
\iff{} & \forall a \in A : (v(a) = f(v)(a))\\
\iff{} & \forall a \in A : (v(a) = \mathbf{t} \leftrightarrow f(v)(a) = \mathbf{t})\\
\iff{} & \forall a \in A : (v(a) = \mathbf{t} \leftrightarrow (\exists w \in \mathcal{V}_2 : f(w)(a) = \mathbf{t}\\
& \qquad\qquad\qquad\qquad\qquad \wedge v = w))\\
\iff{} & \forall a \in A : (v(a) = \mathbf{t} \leftrightarrow (\exists w \in \mathcal{V}_2 : f(w)(a) = \mathbf{t}\\
& \qquad\qquad\qquad\qquad\qquad \wedge v(\phi_w) = \mathbf{t}))\\
\iff{} & \forall a \in A : \left(v(a) = \mathbf{t} \leftrightarrow v\left(\bigvee_{\substack{w \in \mathcal{V}_2,\\ f(w)(a) = \mathbf{t}}} \phi_w \right) = \mathbf{t} \right)\\
\iff{} & \forall a \in A : v(a) = v\left(\bigvee_{\substack{w \in \mathcal{V}_2,\\ f(w)(a) = \mathbf{t}}} \phi_w \right)\\
\iff{} & \forall a \in A : v(a) = v(\varphi_a^f) \iff v \in mod(D_f) \qquad \square
\end{aligned}$$

A related result was given by Strass (2015, Proposition 10). The characterization we presented here fits into the general framework of this paper and is directly usable for our realizability algorithm. Wrapping up, the next result summarizes how ADF realizability can be captured by different types of characterizations for the semantics we considered so far.

Theorem 5. *Let $V \subseteq \mathcal{V}$ be a set of interpretations and consider $\sigma \in \{adm, com, mod\}$. There is an ADF D such that $\sigma(D) = V$ if and only if there is a σ-characterization for V.*

The preferred semantics of an ADF D is closely related to its admissible semantics as, by definition, the preferred interpretations of D are its $\leq_i$-maximal admissible interpretations. As a consequence we can also describe preferred realizability in terms of adm-characterizations. We use the lattice-theoretic standard notation $\max_{\leq_i} V$ to select the $\leq_i$-maximal elements of a given set V of interpretations.

Corollary 6. *Let $V \subseteq \mathcal{V}$ be a set of interpretations. There is an ADF D with $prf(D) = V$ iff there is an adm-characterization for some $V' \subseteq \mathcal{V}$ with $V \subseteq V'$ and $\max_{\leq_i} V' = V$.*

Finally, we give a result on the complexity of deciding realizability for the mentioned formalisms and semantics.

Proposition 7. *Let $\mathcal{F} \in \{AF, SETAF, BADF, ADF\}$ be a formalism and $\sigma \in \{adm, com, prf, mod\}$ be a semantics. The decision problem "Given a vocabulary A and a set $V \subseteq \mathcal{V}$ of interpretations over A, is there a* $\mathsf{kb} \in \mathcal{F}$ *such that $\sigma(\mathsf{kb}) = V$?" can be decided in nondeterministic time that is polynomial in the size of V.*[1]

Proof. For all considered $\mathcal{F}$ and σ, computing all σ-interpretations of a given witness $\mathsf{kb} \in \mathcal{F}$ can be done in time that is linear in the size of V. Comparing the result to V can also be done in linear time. □

3.1 Deciding Realizability: Algorithm 1

Our main algorithm for deciding realizability is a propagate-and-guess algorithm in the spirit of the DPLL algorithm for deciding propositional satisfiability (Gomes et al., 2008). It is generic with respect to (1) the formalism $\mathcal{F}$ and (2) the semantics σ for which should be realized. To this end, the propagation part of the algorithm is kept exchangeable and will vary depending on formalism and semantics. Roughly, in the propagation step the algorithm uses the desired set V of interpretations to derive certain necessary properties of the realizing knowledge base (line 2). This is the essential part of the algorithm: the derivation rules (*propagators*) used there are based on characterizations of realizability with respect to formalism and semantics. Once propagation of properties has reached a fixed point (line 7), the algorithm checks whether the derived information is sufficient to construct a knowledge base. If so, the knowledge base can be constructed and returned (line 9). Otherwise (no more information can be obtained through propagation and there is not enough information to construct a knowledge base yet), the algorithm guesses another assignment for the characterization (line 11) and calls itself recursively.

Algorithm 1 $realize(\mathcal{F}, \sigma, V, F)$

```
Input:  • a formalism $\mathcal{F}$
        • a semantics $\sigma$ for $\mathcal{F}$
        • a set $V$ of interpretations $v : A \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$
        • a relation $F \subseteq \mathcal{V}_2 \times A \times \{\mathbf{t}, \mathbf{f}\}$, initially empty
Output: a $\mathsf{kb} \in \mathcal{F}$ with $\sigma(\mathsf{kb}) = V$ or "no" if none exists
 1: repeat
 2:     set $F_\Delta := \bigcup_{p \in P_\sigma^{\mathcal{F}}} p(V, F) \setminus F$
 3:     set $F := F \cup F_\Delta$
 4:     if $\exists v \in \mathcal{V}_2, \exists a \in A : \{(v, a, \mathbf{t}), (v, a, \mathbf{f})\} \subseteq F$ then
 5:         return "no"
 6:     end if
 7: until $F_\Delta = \emptyset$
 8: if $\forall v \in \mathcal{V}_2, \forall a \in A, \exists x \in \{\mathbf{t}, \mathbf{f}\} : (v, a, x) \in F$ then
 9:     return $kb_\sigma^{\mathcal{F}}(F)$
10: end if
11: choose $v \in \mathcal{V}_2, a \in A$ with $(v, a, \mathbf{t}) \notin F$, $(v, a, \mathbf{f}) \notin F$
12: if $realize(\mathcal{F}, \sigma, V, F \cup \{(v, a, \mathbf{t})\}) \neq$ "no" then
13:     return $realize(\mathcal{F}, \sigma, V, F \cup \{(v, a, \mathbf{t})\})$
14: else
15:     return $realize(\mathcal{F}, \sigma, V, F \cup \{(v, a, \mathbf{f})\})$
16: end if
```

The main data structure that Algorithm 1 operates on is a set of triples $(v, a, \mathbf{x})$ consisting of a two-valued interpretation $v \in \mathcal{V}_2$, an atom $a \in A$ and a truth value $\mathbf{x} \in \{\mathbf{t}, \mathbf{f}\}$. This data structure is intended to represent the σ-characterizations introduced in Definitions 1 to 3. There, a σ-characterization is a function $f : \mathcal{V}_2 \to \mathcal{V}_2$ from two-valued interpretations to two-valued interpretations. However, as the algorithm builds the σ-characterization step by step and there might not even be a σ-characterization in the end (because V is not realizable), we use a set F of triples $(v, a, \mathbf{x})$ to be able to represent both partial and incoherent states of affairs. The σ-characterization candidate induced by F is partial if we have that for some v and a, neither $(v, a, \mathbf{t}) \in F$ nor $(v, a, \mathbf{f}) \in F$; likewise, the candidate is incoherent if for some v and a, both $(v, a, \mathbf{t}) \in F$ and $(v, a, \mathbf{f}) \in F$. If F is neither partial nor incoherent, it gives rise to a unique σ-characterization that can be used to construct the knowledge base realizing the desired set of interpretations. The correspondence to the characterization-function is then such that $f(v)(a) = \mathbf{x}$ iff $(v, a, \mathbf{x}) \in F$.

In our presentation of the algorithm we focused on its main features, therefore the guessing step (line 11) is completely "blind". It is possible to use common CSP techniques, such as shaving (removing guessing possibilities that directly lead to inconsistency). Finally, we remark that the algorithm can be extended to enumerate all possible realizations of a given interpretation set – by keeping all choice points in the guessing step and thus exhaustively exploring the whole search space.

In the case where the constructed relation F becomes functional at some point, the algorithm returns a realizing knowledge base $kb_\sigma^{\mathcal{F}}(F)$. For ADFs, this just means that we denote by f the σ-characterization represented by F and set $kb_\sigma^{\mathrm{ADF}}(F) = D^f$. For the remaining formalisms we will introduce the respective constructions in later subsections.

The algorithm is parametric in two dimensions, namely with respect to the formalism $\mathcal{F}$ and with respect to the semantics σ. These two aspects come into the algorithm via

[1] We assume here that the representation of any V over A has size $\Theta(3^{|A|})$. There might be specific V with smaller representations, but we cannot assume any better for the general case.

$$\begin{aligned}
p^{\in}_{adm}(V,F) &= \{(v_2, a, v(a)) \mid v \in V, v_2 \in [v]_2, v(a) \neq \mathbf{u}\}\\
p^{\notin}_{adm}(V,F) &= \{(v_2, a, \neg v(a)) \mid v \in \mathcal{V} \setminus V, v_2 \in [v]_2,\\
&\qquad v(a) \neq \mathbf{u}, \forall b \in A \setminus v^{-1}(\mathbf{u}), \forall v_2' \in [v]_2 :\\
&\qquad (a, v_2) \neq (b, v_2') \rightarrow (v_2', b, v(b)) \in F\}\\
p^{\lightning}_{adm}(V,F) &= \{(v, a, \mathbf{t}), (v, a, \mathbf{f}) \mid v \in \mathcal{V}_2, a \in A, v_{\mathbf{u}} \notin V\}\\
p^{\in}_{mod}(V,F) &= \{(v, a, v(a)) \mid v \in V, a \in A\}\\
p^{\notin}_{mod}(V,F) &= \{(v, a, \neg v(a)) \mid v \in \mathcal{V}_2 \setminus V, a \in A,\\
&\qquad \forall c \in A \setminus \{a\} : (v, c, v(c)) \in F\}\\
p^{\lightning}_{mod}(V,F) &= \{(v, a, \mathbf{t}), (v, a, \mathbf{f}) \mid v \in \mathcal{V}_2, a \in A, V \not\subseteq \mathcal{V}_2\}
\end{aligned}$$

$$\begin{aligned}
p^{\in,\mathbf{u}}_{com}(V,F) &= \{(v_2, a, \neg\mathbf{x}) \mid v \in V, v_2 \in [v]_2, v(a) = \mathbf{u},\\
&\qquad \mathbf{x} \in \{\mathbf{t},\mathbf{f}\}, \forall v_2' \in [v]_2 : v_2 \neq v_2' \rightarrow (v_2', a, \mathbf{x}) \in F\}\\
p^{\notin,\mathbf{tf}}_{com}(V,F) &= \{(v_2, a, \neg v(a)) \mid v \in \mathcal{V} \setminus V, v_2 \in [v]_2, v(a) \neq \mathbf{u},\\
&\qquad \forall b \in A \setminus v^{-1}(\mathbf{u}), \forall v_2' \in [v]_2 : (a, v_2) \neq (b, v_2') \rightarrow (v_2', b, v(b)) \in F,\\
&\qquad \forall b \in v^{-1}(\mathbf{u}), \exists v_2'', v_2''' \in [v]_2 : (v_2'', b, \mathbf{t}), (v_2''', b, \mathbf{f}) \in F\}\\
p^{\notin,\mathbf{u}}_{com}(V,F) &= \{(v_2, a, \neg\mathbf{x}) \mid v \in \mathcal{V} \setminus V, v_2 \in [v]_2, v(a) = \mathbf{u},\\
&\qquad \forall b \in A \setminus v^{-1}(\mathbf{u}), \forall v_2' \in [v]_2 : (v_2, b, v(b)) \in F,\\
&\qquad \forall b \in v^{-1}(\mathbf{u}) \setminus \{a\} : \exists v_2'', v_2''' \in [v]_2 : (v_2'', b, \mathbf{t}),\\
&\qquad (v_2''', b, \mathbf{f}) \in F, \forall v_2'''' \in [v]_2 \setminus \{v_2\} : (v_2'''', b, \mathbf{x}) \in F\}
\end{aligned}$$

Figure 1: Semantics propagators for the complete ($P^{ADF}_{com} = \{p^{\in,\mathbf{tf}}_{com}, p^{\in,\mathbf{u}}_{com}, p^{\notin,\mathbf{tf}}_{com}, p^{\notin,\mathbf{u}}_{com}\}$ with $p^{\in,\mathbf{tf}}_{com}(V,F) = p^{\in}_{adm}(V,F)$), admissible ($P^{ADF}_{adm} = \{p^{\in}_{adm}, p^{\notin}_{adm}, p^{\lightning}_{adm}\}$), and model semantics ($P^{ADF}_{mod} = \{p^{\in}_{mod}, p^{\notin}_{mod}, p^{\lightning}_{mod}\}$).

so-called *propagators*. A propagator is a formalism-specific or semantics-specific set of derivation rules. Given a set V of desired interpretations and a partial σ-characterization F, a propagator p derives new triples $(v, a, \mathbf{x})$ that must necessarily be part of any total σ-characterization f for V such that f extends F. In the following, we present semantics propagators for admissible, complete and two-valued model (in (SET)AF terms stable) semantics, and formalism propagators for BADFs, AFs, and SETAFs.

3.2 Semantics Propagators

These propagators (cf. Figure 1) are directly derived from the properties of σ-characterizations presented in Definitions 1 to 3. While the definitions provide exact conditions to check whether a given function is a σ-characterization, the propagators allow us to derive definite values of partial characterizations that are necessary to fulfill the conditions for being a σ-characterization.

For admissible semantics, the condition for a function f to be an adm-characterization of a desired set of interpretations V (cf. Definition 2) can be split into a condition for desired interpretations $v \in V$ and two conditions for undesired interpretations $v \notin V$. Propagator $p^{\in}_{adm}$ derives new triples by considering interpretations $v \in V$. Here, for all two-valued interpretations v_2 that extend v, the value $f(v_2)$ has to be in accordance with v on v's Boolean part, that is, the algorithm adds $(v_2, a, v(a))$ whenever $v(a) \neq \mathbf{u}$. On the other hand, $p^{\notin}_{adm}$ derives new triples for $v \notin V$ in order to ensure that there is a two-valued interpretation v_2 extending v where $f(v_2)$ differs from v on a Boolean value of v. Note that while $p^{\in}_{adm}$ immediately allows us to derive information about F for each desired interpretation $v \in V$, propagator $p^{\notin}_{adm}$ is much weaker in the sense that it only derives a triple of F if there is no other way to meet the conditions for an undesired interpretation. Special treatment is required for the interpretation $v_{\mathbf{u}}$ that maps all statements to $\mathbf{u}$ and is admissible for every ADF. This is not captured by $p^{\in}_{adm}$ and $p^{\notin}_{adm}$ as these deal only with interpretations that have Boolean mappings. Thus, propagator $p^{\lightning}_{adm}$ serves to check whether $v_{\mathbf{u}} \in V$. If this is not the case, the propagator immediately makes the relation F incoherent and the algorithm correctly answers "no".

For complete semantics and interpretations $v \in V$, propagator $p^{\in,\mathbf{tf}}_{com}$ derives triples just like in the admissible case. Propagator $p^{\in,\mathbf{u}}_{com}$ deals with statements $a \in A$ having $v(a) = \mathbf{u}$ for which there have to be at least two $v_2, v_2' \in [v]_2$ having $f(v_2)(a) = \mathbf{t}$ and $f(v_2')(a) = \mathbf{f}$. Hence $p^{\in,\mathbf{u}}_{com}$ derives triple $(v_2, a, \neg\mathbf{x})$ if for all other $v_2' \in [v]_2$ we find a triple $(v_2', a, \mathbf{x})$. For interpretations $v \notin V$ it must hold that there is some $a \in A$ such that (i) $v(a) \neq \mathbf{u}$ and $f(v_2)(a) \neq v(a)$ for some $v_2 \in [v]_2$ or (ii) $v(a) = \mathbf{u}$ but for all $v_2 \in [v]_2$, $f(v_2)$ assigns the same Boolean truth value $\mathbf{x}$ to a. Now if neither (i) nor (ii) can be fulfilled by any statement $b \in A \setminus \{a\}$ due to the current contents of F, propagators $p^{\notin,\mathbf{tf}}_{com}$ and $p^{\notin,\mathbf{u}}_{com}$ derive triple $(v_2, a, \neg v(a))$ for $v(a) \neq \mathbf{u}$ if needed for a to fulfill (i) and $(v_2, a, \neg\mathbf{x})$ for $v(a) = \mathbf{u}$ if needed for a to fulfill (ii), respectively.

Example 2. Consider the set $V_3 = \{\mathbf{uuu}, \mathbf{fuu}, \mathbf{uuf}, \mathbf{ftf}\}$. First, we consider a run of *realize*(ADF, adm, V_3, $\emptyset$). In the first iteration, propagator $p^{\in}_{adm}$ ensures that F_{Δ} in line 2 contains $(\mathbf{fff}, a, \mathbf{f})$, $(\mathbf{ftf}, a, \mathbf{f})$, $(\mathbf{ftf}, c, \mathbf{f})$, and $(\mathbf{fff}, c, \mathbf{f})$. Based on the latter three tuples and $\mathbf{fuf} \notin V_3$, propagator $p^{\notin}_{adm}$ derives $(\mathbf{fff}, a, \mathbf{t})$ in the second iteration which together with $(\mathbf{fff}, a, \mathbf{f})$ causes the algorithm to return "no". Consequently, V_3 is not adm-realizable. A run of *realize*(ADF, com, V_3, $\emptyset$) on the other hand returns com-characterization f for V_3 that maps $\mathbf{ttf}$ to $\mathbf{tff}$, $\mathbf{ftt}$ to $\mathbf{fft}$, $\mathbf{ftf}$ and $\mathbf{fff}$ to $\mathbf{ftf}$ and all other $v_2 \in \mathcal{V}_2$ to $\mathbf{fff}$. Hence, ADF D_f, given by the acceptance conditions

$$\varphi^f_a = a \wedge b \wedge \neg c, \qquad \varphi^f_c = \neg a \wedge b \wedge c,$$
$$\varphi^f_b = (\neg a \wedge b \neg \wedge \neg c) \vee (\neg a \wedge \neg b \wedge \neg c)$$

has V_3 as its complete semantics. ■

Finally, for two-valued model semantics, propagator $p^{\in}_{mod}$ derives new triples by looking at interpretations $v \in V$. For those, we must find $f(v) = v$ in each mod-characterization f by definition. Thus the algorithm adds $(v, a, v(a))$ for each $a \in A$ to the partial characterization F. Propagator $p^{\notin}_{mod}$ looks at interpretations $v \in \mathcal{V}_2 \setminus V$, for which it must hold that $f(v) \neq v$. Thus there must be a statement $a \in A$ with $v(a) \neq f(v)(a)$, which is exactly what this propagator

Algorithm 2 $\mathit{realizePrf}(\mathcal{F}, V)$

```
Input:  • a formalism $\mathcal{F}$
        • a set $V$ of interpretations $v : A \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$
Output: Return some $\mathsf{kb} \in \mathcal{F}$ with $\mathit{prf}(\mathsf{kb}) = V$ if one
        exists or “no” otherwise.
 1: if $\max_{\leq_i} V \neq V$ then
 2:     return “no”
 3: end if
 4: set $V^{<} := \{v \in \mathcal{V} \mid \exists v' \in V : v <_i v'\}$
 5: set $X := \emptyset$
 6: repeat
 7:     choose $V' \subseteq V^{<}$ with $V' \notin X$
 8:     set $X := X \cup \{V'\}$
 9:     set $V^{adm} := V \cup V'$
10:     if $\mathit{realize}(\mathcal{F}, \mathit{adm}, V^{adm}, \emptyset) \neq$ “no” then
11:         return $\mathit{realize}(\mathcal{F}, \mathit{adm}, V^{adm}, \emptyset)$
12:     end if
13: until $\forall V' \subseteq V^{<} : V' \in X$
14: return “no”
```

derives whenever it is clear that there is only one statement candidate left. This, in turn, is the case whenever all $b \in A$ with the opposite truth value $\neg v(a)$ and all $c \in A$ with $c \neq a$ cannot coherently become the necessary witness any more. The propagator $p^{\sharp}_{mod}$ checks whether $V \subseteq \mathcal{V}_2$, that is, the desired set of interpretations consists entirely of two-valued interpretations. In that case this propagator makes the relation F incoherent, following a similar strategy as $p^{\sharp}_{adm}$.

Preferred Semantics Realizing a given set of interpretations V under preferred semantics requires special treatment. We do not have a σ-characterization function for $\sigma = \mathit{prf}$ at hand to directly check realizability of V but have to find some $V' \subseteq \{v \in \mathcal{V} \mid \exists v' \in V : v <_i v'\}$ such that $V \cup V'$ is realizable under admissible semantics (cf. Corollary 6). Algorithm 2 implements this idea by guessing such a V' (line 7) and then using Algorithm 1 to try to realize $V \cup V'$ under admissible semantics (line 11). If *realize* returns a knowledge base kb realizing $V \cup V'$ under *adm* we can directly use kb as solution of *realizePrf* since it holds that $\mathit{prf}(\mathsf{kb}) = V$, given that V is an $\leq_i$-antichain (line 2).

3.3 Formalism Propagators

When constructing an ADF realizing a given set V of interpretations under a semantics σ, the function $kb^{\text{ADF}}_{\sigma}(F)$ makes use of the σ-characterization given by F in the following way: v is a model of the acceptance condition φ_a if and only if we find $(v, a, \mathbf{t}) \in F$. Now as bipolar ADFs, SETAFs and AFs are all subclasses of ADFs by restricting the acceptance conditions of statements, these restrictions also carry over to the σ-characterizations. The propagators defined below use structural knowledge on the form of acceptance conditions of the respective formalisms to reduce the search space or to induce incoherence of F whenever V is not realizable.

Bipolar ADFs For bipolar ADFs, we use the fact that each of their links must have at least one polarity, that is, must be supporting or attacking. Therefore, if a link is not supporting, it must be attacking, and vice versa. For canonical realization, we obtain the polarities of links, i.e. the sets L^+ and L^-, as defined in Figure 2.

AFs To explain the AF propagators, we first need some more definitions. On the two classical truth values, we define the truth ordering $\mathbf{f} <_t \mathbf{t}$, whence the operations $\sqcup_t$ and $\sqcap_t$ with $\mathbf{f} \sqcup_t \mathbf{t} = \mathbf{t}$ and $\mathbf{f} \sqcap_t \mathbf{t} = \mathbf{f}$ result. These operations can be lifted pointwise to two-valued interpretations as usual, that is, $(v_1 \sqcup_t v_2)(a) = v_1(a) \sqcup_t v_2(a)$ and $(v_1 \sqcap_t v_2)(a) = v_1(a) \sqcap_t v_2(a)$. Again, the reflexive version of $<_t$ is denoted by $\leq_t$. The pair $(\mathcal{V}_2, \leq_t)$ of two-valued interpretations ordered by the truth ordering forms a complete lattice with glb $\sqcap_t$ and lub $\sqcup_t$. This complete lattice has the least element $v_{\mathbf{f}} : A \to \{\mathbf{f}\}$, the interpretation mapping all statements to false, and the greatest element $v_{\mathbf{t}} : A \to \{\mathbf{t}\}$ mapping all statements to true, respectively.

Acceptance conditions of AF-based ADFs have the form of conjunctions of negative literals. In the complete lattice $(\mathcal{V}_2, \leq_t)$, the model sets of AF acceptance conditions correspond to the lattice-theoretic concept of an *ideal*, a subset of $\mathcal{V}_2$ that is downward-closed with respect to $\leq_t$ and upward-closed with respect to $\sqcup_t$. The propagator directly implements these closure properties: application of p^{AF} ensures that when a σ-characterization F that is neither incoherent nor partial is found in line 8 of Algorithm 1, then there is, for each $a \in A$, an interpretation v_a such that $(v_a, a, \mathbf{t}) \in F$ and $v \leq_t v_a$ for each $(v, a, \mathbf{t}) \in F$. Hence v_a is crucial for the acceptance condition, or in AF terms the attacks, of a and we can define $kb^{\text{AF}}_{\sigma}(F) = (A, \{(b, a) \mid a, b \in A, v_a(b) = \mathbf{f}\})$.

SETAFs The propagator for SETAFs, p^{SETAF}, is a weaker version of that of AFs, since we cannot presume upward-closure with respect to $\sqcup_t$. In SETAF-based ADFs the acceptance formula is in *conjunctive normal form* containing only negative literals. By a transformation preserving logical equivalence we obtain an acceptance condition in *disjunctive normal form*, again with only negative literals; in other words, a *disjunction* of AF acceptance formulas. Thus, the model set of a SETAF acceptance condition is not necessarily an ideal, but a union of ideals. For the canonical realization we can make use of the fact that, for each $a \in A$, the set $V^{\mathbf{t}}_a = \{v \in \mathcal{V}_2 \mid (v, a, \mathbf{t}) \in F\}$ is downward-closed with respect to $\leq_t$, hence the set of models of $\bigvee_{v \in \max_{\leq_t} V^{\mathbf{t}}} \bigwedge_{v(b)=\mathbf{f}} \neg b$ is exactly $V^{\mathbf{t}}_a$. The clauses of its corresponding CNF-formula exactly coincide with the sets of arguments attacking a in $kb^{\text{SETAF}}_{\sigma}(F)$.

3.4 Correctness

For a lack of space, we could not include a formal proof of soundness and completeness of Algorithm 1, but rather present arguments for termination and correctness.

Termination With each recursive call, the set F can never decrease in size, as the only changes to F are adding the results of propagation in line 3 and adding the guesses in line 11. Also within the until-loop, the set F can never decrease in size; furthermore there is only an overall finite number of triples that can be added to F. Thus at some point

$$p^{\text{SETAF}}(V,F) = \{(v_{\mathbf{f}}, a, \mathbf{t}) \mid a \in A\} \cup \{(w,a,\mathbf{t}) \mid (v,a,\mathbf{t}) \in F, w \in \mathcal{V}_2, w <_t v\} \cup \{(w,a,\mathbf{f}) \mid (v,a,\mathbf{f}) \in F, w \in \mathcal{V}_2, v <_t w\}$$

$$p^{\text{AF}}(V,F) = p^{\text{SETAF}}(V,F) \cup \{(v_1 \sqcup_t v_2, a, \mathbf{t}) \mid (v_1,a,\mathbf{t}) \in F, (v_2,a,\mathbf{t}) \in F\} \qquad L^+ = \left\{(b,a) \bigm| (v,a,\mathbf{f}) \in F, v(b) = \mathbf{f}, (v|^b_{\mathbf{t}}, a, \mathbf{t}) \in F\right\}$$

$$p^{\text{BADF}}(V,F) = \{(v|^b_{\mathbf{t}}, a, \mathbf{x}) \mid (v,a,\mathbf{x}) \in F, (w,a,\neg\mathbf{x}) \in F, w(b) = \mathbf{f}, (w|^b_{\mathbf{t}}, a, \mathbf{x}) \in F\} \qquad L^- = \left\{(b,a) \bigm| (v,a,\mathbf{t}) \in F, v(b) = \mathbf{f}, (v|^b_{\mathbf{t}}, a, \mathbf{f}) \in F\right\}$$

Figure 2: Formalism propagators. For formalism $\mathcal{F} \in \{AF, SETAF, BADF\}$ and any $\sigma \in \{adm, com, prf, mod\}$, we set the respective propagator for $\mathcal{F}$ to $P^{\mathcal{F}}_\sigma = P^{ADF}_\sigma \cup \{p^{\mathcal{F}}\}$ with $p^{\mathcal{F}}$ as defined above. L^+ and L^- define link polarities for kb^{BADF}_σ.

we must have $F_\Delta = \emptyset$ and leave the until-loop. Since F always increases in size, at some point it must either become functional or incoherent, whence the algorithm terminates.

Soundness If the algorithm returns a realizing knowledge base $kb^{\mathcal{F}}_\sigma(F)$, then according to the condition in line 8 the relation F induced a total function $f : \mathcal{V}_2 \to \mathcal{V}_2$. In particular, because the until-loop must have been run through at least once, there was at least one propagation step (line 2). Since the propagators are defined such that they enforce everything that must hold in a σ-characterization, we conclude that the induced function f indeed is a σ-characterization for V. By construction, we consequently find that $\sigma(kb^{\mathcal{F}}_\sigma(F)) = V$.

Completeness If the algorithm answers "no", then the execution reached line 5. Thus, for the constructed set F, there must have been an interpretation $v \in \mathcal{V}_2$ and a statement $a \in A$ such that $\{(v,a,\mathbf{t}), (v,a,\mathbf{f})\} \subseteq F$, that is, F is incoherent. Since F is initially empty, the only way it could get incoherent is in the propagation step in line 2. (The guessing step cannot create incoherence, since exactly one truth value is guessed for v and a.) However, the propagators are defined such that they infer only assignments (triples) that are necessary for the given F. Consequently, the given interpretation set V is such that either there is no realization within the ADF fragment corresponding to formalism $\mathcal{F}$ (that is, the formalism propagator derived the incoherence) or there is no σ-characterization for V with respect to general ADFs (that is, the semantics propagator derived the incoherence). In any case, V is not σ-realizable for $\mathcal{F}$.

4 Implementation

As Algorithm 1 is based on propagation, guessing, and checking it is perfectly suited for an implementation using answer set programming (ASP) (Niemelä, 1999; Marek and Truszczyński, 1999) as this allows for exploiting conflict learning strategies and heuristics of modern ASP solvers. Thus, we developed ASP encodings in the `Gringo` language (Gebser et al., 2012) for our approach. Similar as the algorithm, our declarative encodings are modular, consisting of a main part responsible for constructing set F and separate encodings for the individual propagators. If one wants, e.g., to compute an AF realization under admissible semantics for a set V of interpretations, an input program encoding V is joined with the main encoding, the propagator encoding for admissible semantics as well as the propagator encoding for AFs. Every answer set of such a program encodes a respective characterization function. Our ASP encoding for preferred semantics is based on the admissible encoding and guesses further interpretations following the essential idea of Algorithm 2. For constructing a knowledge base with the desired semantics, we also provide two ASP encodings that transform the output to an ADF in the syntax of the `DIAMOND` tool (Ellmauthaler and Strass, 2014), respectively an AF in `ASPARTIX` syntax (Egly, Gaggl, and Woltran, 2010; Gaggl et al., 2015). Both argumentation tools are based on ASP themselves. The encodings for all the semantics and formalisms we covered in the paper can be downloaded from `http://www.dbai.tuwien.ac.at/research/project/adf/unreal/`. A selection of them is depicted in Figure 3 on the next page.

5 Expressiveness Results

In this section we briefly present some results that we have obtained using our implementation. We first introduce some necessary notation to describe the relative expressiveness of knowledge representation formalisms (Gogic et al., 1995; Strass, 2015). For formalisms $\mathcal{F}_1$ and $\mathcal{F}_2$ with semantics σ_1 and σ_2, we say that $\mathcal{F}_2$ under σ_2 is at least as expressive as $\mathcal{F}_1$ under σ_1 and write $\mathcal{F}_1^{\sigma_1} \leq_e \mathcal{F}_2^{\sigma_2}$ if and only if $\Sigma^{\sigma_1}_{\mathcal{F}_1} \subseteq \Sigma^{\sigma_2}_{\mathcal{F}_2}$, where $\Sigma^\sigma_{\mathcal{F}} = \{\sigma(\mathsf{kb}) \mid \mathsf{kb} \in \mathcal{F}\}$ is the *signature of $\mathcal{F}$ under σ*. As usual, we define $\mathcal{F}_1 <_e \mathcal{F}_2$ iff $F_1 \leq_e \mathcal{F}_2$ and $F_2 \not\leq_e \mathcal{F}_1$.

We now start by considering the signatures of AFs, SETAFs and (B)ADFs for the unary vocabulary $\{a\}$:

$$\begin{aligned}
\Sigma^{adm}_{\text{AF}} &= \Sigma^{adm}_{\text{SETAF}} = \{\{\mathbf{u}\}, \{\mathbf{u},\mathbf{t}\}\} \\
\Sigma^{com}_{\text{AF}} &= \Sigma^{com}_{\text{SETAF}} = \{\{\mathbf{u}\}, \{\mathbf{t}\}\} \\
\Sigma^{prf}_{\text{AF}} &= \Sigma^{prf}_{\text{SETAF}} = \{\{\mathbf{u}\}, \{\mathbf{t}\}\} \\
\Sigma^{mod}_{\text{AF}} &= \Sigma^{mod}_{\text{SETAF}} = \{\emptyset, \{\mathbf{t}\}\} \\
\Sigma^{adm}_{\text{ADF}} &= \Sigma^{adm}_{\text{BADF}} = \Sigma^{adm}_{\text{AF}} \cup \{\{\mathbf{u},\mathbf{f}\}, \{\mathbf{u},\mathbf{t},\mathbf{f}\}\} \\
\Sigma^{com}_{\text{ADF}} &= \Sigma^{com}_{\text{BADF}} = \Sigma^{com}_{\text{AF}} \cup \{\{\mathbf{f}\}, \{\mathbf{u},\mathbf{t},\mathbf{f}\}\} \\
\Sigma^{prf}_{\text{ADF}} &= \Sigma^{prf}_{\text{BADF}} = \Sigma^{prf}_{\text{AF}} \cup \{\{\mathbf{f}\}, \{\mathbf{t},\mathbf{f}\}\} \\
\Sigma^{mod}_{\text{ADF}} &= \Sigma^{mod}_{\text{BADF}} = \Sigma^{mod}_{\text{AF}} \cup \{\{\mathbf{f}\}, \{\mathbf{t},\mathbf{f}\}\}
\end{aligned}$$

The following result shows that the expressiveness of the formalisms under consideration is in line with the amount of restrictions they impose on acceptance formulas.

Theorem 8. *For any $\sigma \in \{adm, com, prf, mod\}$:*

1. *$AF^\sigma <_e SETAF^\sigma$.*
2. *$SETAF^\sigma <_e BADF^\sigma$.*
3. *$BADF^\sigma <_e ADF^\sigma$.*

Proof. (1) $\text{AF}^\sigma \leq_e \text{SETAF}^\sigma$ is clear (by modeling individual attacks via singletons). For $\text{SETAF}^\sigma \not\leq_e \text{AF}^\sigma$ the witnessing model sets over vocabulary $A = \{a,b,c\}$ are $\{\mathbf{uuu}, \mathbf{ttf}, \mathbf{tft}, \mathbf{ftt}\} \in \Sigma^\sigma_{\text{SETAF}} \setminus \Sigma^\sigma_{\text{AF}}$ and $\{\mathbf{ttf}, \mathbf{tft}, \mathbf{ftt}\} \in$

```
Main Encoding
% s/1 declares statements
% a cterm assigns a truth value to a statement
cterm(A, t(A)) :- s(A).
cterm(A, f(A)) :- s(A).

% int/1 declares interpretations
% an interpretation is an ordered list of assignments
% of statements to one of two truth values
% unassigned statements have truth value u
int(nil).
int((AS, I)) :- s(A), cterm(A, AS),
             int(I), smaller(A, I).
smaller(A, nil) :- s(A).
smaller(A, (H, I)) :- s(A), cterm(T, H), A < T, int(I).

% check whether an interpretation contains an assignment
member(T, (T, I)) :- int((T, I)).
member(T, (X, I)) :- int((X, I)), member(T, I).

% an interpretation is two-valued
int2(I) :- int(I), not hasU(I).
hasU(I) :- hasU(I, A).
hasU(I, A) :- int(I), s(A), not member(t(A), I),
                            not member(f(A), I).

% the information ordering on interpretations
ileq(I, J) :- int(I), int(J), not nileq(I, J).
nileq(I, J) :- int(I), int(J), member(T, I),
                           not member(T, J).

% guess a characterization function
1 { ch(A, I, t); ch(A, I, f) } 1 :- s(A), int2(I).
```

```
Two-Valued Model Encoding
% only sets of two-valued interpretations realizable
:- in(I), not int2(I).

% propagation members
ch(A, I, t) :- int2(I), in(I), s(A), member(t(A), I).
ch(A, I, f) :- int2(I), in(I), s(A), member(f(A), I).

% propagation non-members
ch(A, I, t) :- int2(I), not in(I), member(f(A), I),
                ch(B, I, t) : s(B), member(t(B), I);
        ch(C, I, f) : s(C), member(f(C), I), C != A.
ch(A, I, f) :- int2(I), not in(I), member(t(A), I),
                ch(B, I, f) : s(B), member(f(B), I);
        ch(C, I, t) : s(C), member(t(C), I), C != A.
```

```
BADF Encoding
% if a statement sometimes attacks, it cannot be
% supporting, therefore must be attacking
att(B, A) :- ch(A, I, t), ch(A, J, f), diffFT(I, J, B).

% if a statement sometimes supports, it cannot be
% attacking, therefore it must be supporting
sup(B, A) :- ch(A, I, f), ch(A, J, t), diffFT(I, J, B).

% derives two-valued I and J that differ only for A
diffFT(I, J, A) :- int2(I), int2(J), member(f(A), I, B),
                                     member(t(A), J, B).
member(T, (T, I), I) :- int((T, I)).
member(T, (X, I), (X, B)) :- int((X, I)),
                  member(T, I, B), X != T.

% if a statement is supporting/attacking for another,
% then this information can be propagated
ch(A, J, f) :- att(B, A), ch(A, I, f), diffFT(I, J, B).
ch(A, J, t) :- sup(B, A), ch(A, I, t), diffFT(I, J, B).
```

Figure 3: Selected ASP encodings in clingo 4 syntax. The main encoding implements Algorithm 1, the remaining encodings implement the two-valued model semantics propagator, and the BADF formalism propagator, respectively.

$\Sigma^{\tau}_{\text{SETAF}} \setminus \Sigma^{\tau}_{\text{AF}}$ with $\sigma \in \{adm, com\}$ and $\tau \in \{prf, mod\}$. By each pair of arguments of A being $\mathbf{t}$ in at least one model, a realizing AF cannot feature any attack, immediately giving rise to the model $\mathbf{ttt}$. The respective realizing SETAF is given by the attack relation $R = \{(\{a,b\},c), (\{a,c\},b), (\{b,c\},a)\}$.

(2) It is clear that SETAF$^\sigma \leq_e$ BADF$^\sigma$ holds (all parents are always attacking). For BADF$^\sigma \not\leq_e$ SETAF$^\sigma$ the respective counterexamples can be read off the signatures above: for $\sigma \in \{adm, com\}$ we find $\{\mathbf{u}, \mathbf{t}, \mathbf{f}\} \in \Sigma^{\sigma}_{\text{BADF}} \setminus \Sigma^{\sigma}_{\text{SETAF}}$ and for $\tau \in \{prf, mod\}$ we find $\{\mathbf{t}, \mathbf{f}\} \in \Sigma^{\tau}_{\text{BADF}} \setminus \Sigma^{\tau}_{\text{SETAF}}$.

(3) For $\sigma = mod$ the result is known (Strass, 2015, Theorem 14); for the remaining semantics the model sets witnessing ADF$^\sigma \not\leq_e$ BADF$^\sigma$ over vocabulary $A = \{a, b\}$ are

$$\{\mathbf{uu}, \mathbf{tu}, \mathbf{tt}, \mathbf{tf}, \mathbf{fu}\} \in \Sigma^{adm}_{\text{ADF}} \setminus \Sigma^{adm}_{\text{BADF}}$$
$$\{\mathbf{uu}, \mathbf{tu}, \mathbf{tt}, \mathbf{tf}, \mathbf{fu}\} \in \Sigma^{com}_{\text{ADF}} \setminus \Sigma^{com}_{\text{BADF}}$$
$$\{\mathbf{tt}, \mathbf{tf}, \mathbf{fu}\} \in \Sigma^{prf}_{\text{ADF}} \setminus \Sigma^{prf}_{\text{BADF}}$$

A witnessing ADF is given by $\varphi_a = a$ and $\varphi_b = a \leftrightarrow b$. □

Theorem 8 is concerned with the relative expressiveness of the formalisms under consideration, given a certain semantics. Considering different semantics we find that for all formalisms the signatures become incomparable:

Proposition 9. *$\mathcal{F}_1^{\sigma_1} \not\leq_e \mathcal{F}_2^{\sigma_2}$ and $\mathcal{F}_2^{\sigma_2} \not\leq_e \mathcal{F}_1^{\sigma_1}$ for all formalisms $\mathcal{F}_1, \mathcal{F}_2 \in \{AF, SETAF, BADF, ADF\}$ and all semantics $\sigma_1, \sigma_2 \in \{adm, com, prf, mod\}$ with $\sigma_1 \neq \sigma_2$.*

Proof. First, the result for adm and com follows by $\{\mathbf{u}, \mathbf{t}\} \in \Sigma^{adm}_{\text{AF}}$, but $\{\mathbf{u}, \mathbf{t}\} \notin \Sigma^{com}_{\text{ADF}}$ and $\{\mathbf{t}\} \in \Sigma^{com}_{\text{AF}}$, but $\{\mathbf{t}\} \notin \Sigma^{adm}_{\text{ADF}}$. Moreover, taking into account that the set of preferred interpretations (resp. two-valued models) always forms a $\leq_i$-antichain while the set of admissible (resp. complete) interpretations never does, the result follows for $\sigma_1 \in \{adm, com\}$ and $\sigma_2 \in \{prf, mod\}$. Finally, since a kb $\in \mathcal{F}$ may not have any two-valued models and a preferred interpretation is not necessarily two-valued, the result for prf and mod follows. □

Disregarding the possibility of realizing the empty set of interpretations under the two-valued model semantics, we obtain the following relation for ADFs.

Proposition 10. $(\Sigma^{mod}_{ADF} \setminus \{\emptyset\}) \subseteq \Sigma^{prf}_{ADF}$.

Proof. Consider some $V \in \Sigma^{mod}_{\text{ADF}}$ with $V \neq \emptyset$. Clearly $V \subseteq \mathcal{V}_2$ and by Proposition 4 there is a *mod*-characterization $f : \mathcal{V}_2 \to \mathcal{V}_2$ for V, that is, $f(v) = v$ iff $v \in V$. Define $f' : \mathcal{V}_2 \to \mathcal{V}_2$ such that $f'(v) = f(v) = v$ for all $v \in V$ and $f'(v)(a) = \neg v(a)$ for all $v \in \mathcal{V} \setminus V$ and $a \in A$. Now it holds that f' is an *adm*-characterization of $V' = \{v \in \mathcal{V} \mid \forall v_2 \in [v]_2 : v_2 \in V\} \cup \{v_{\mathbf{u}}\}$. Since $\max_{\leq_i} V' = V$ we get that the ADF D with acceptance formula $\varphi_a^{f'}$ for each $a \in A$ has $prf(D) = V$ whence $V \in \Sigma^{prf}_{\text{ADF}}$. □

In contrast, this relation does not hold for AFs, which was shown for extension-based semantics by Linsbichler, Spanring, and Woltran (2015) (Theorem 5) and immediately follows for the three-valued case.

6 Discussion

We presented a framework for realizability in which AFs, SETAFs, BADFs and general ADFs can be treated in a uniform way. The centerpiece of our approach is an algorithm for deciding realizability of a given interpretation-set in a formalism under a semantics. The algorithm makes use of so-called propagators, by which it can be adapted to the different formalisms and semantics. We also presented an implementation of our framework in answer set programming and several novel expressiveness results that we obtained using our implementation. In related work, Polberg (2016) studies a wide range of abstract argumentation formalisms, in particular their relationship with ADFs. This can be the basis for including further formalisms into our realizability framework: all that remains to do is figuring out suitable ADF fragments and developing propagators for them, just like we did exemplarily for Nielsen and Parsons's SETAFs. For further future work, we could also streamline existing propagators such that they do not only derive absolutely necessary assignments, but also logically weaker conclusions, such as disjunctions of (non-)assignments.

References

Amgoud, L., and Cayrol, C. 2002. A reasoning model based on the production of acceptable arguments. *Ann. Math. Artif. Intell.* 34(1–3):197–215.

Baroni, P.; Cerutti, F.; Giacomin, M.; and Guida, G. 2011. AFRA: Argumentation framework with recursive attacks. *Int. J. Approx. Reasoning* 52(1):19–37.

Baumann, R.; Dvořák, W.; Linsbichler, T.; Strass, H.; and Woltran, S. 2014. Compact argumentation frameworks. In *Proc. ECAI*, volume 263 of *FAIA*, 69–74.

Brewka, G., and Woltran, S. 2010. Abstract Dialectical Frameworks. In *Proc. KR*, 102–111.

Brewka, G.; Ellmauthaler, S.; Strass, H.; Wallner, J. P.; and Woltran, S. 2013. Abstract Dialectical Frameworks Revisited. In *Proc. IJCAI*, 803–809.

Brewka, G.; Polberg, S.; and Woltran, S. 2014. Generalizations of Dung frameworks and their role in formal argumentation. *IEEE Intelligent Systems* 29(1):30–38.

Caminada, M., and Gabbay, D. 2009. A logical account of formal argumentation. *Studia Logica* 93(2-3):109–145.

Cayrol, C., and Lagasquie-Schiex, M. 2005. On the acceptability of arguments in bipolar argumentation frameworks. In *Proc. ECSQARU*, volume 3571 of *LNCS*, 378–389.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.* 77(2):321–357.

Dunne, P. E.; Dvořák, W.; Linsbichler, T.; and Woltran, S. 2013. Characteristics of multiple viewpoints in abstract argumentation. In *Proc. DKB*, 16–30.

Dunne, P. E.; Dvořák, W.; Linsbichler, T.; and Woltran, S. 2015. Characteristics of multiple viewpoints in abstract argumentation. *Artif. Intell.* 228:153–178.

Dyrkolbotn, S. K. 2014. How to Argue for Anything: Enforcing Arbitrary Sets of Labellings using AFs. In *Proc. KR*, 626–629.

Egly, U.; Gaggl, S. A.; and Woltran, S. 2010. Answer-set programming encodings for argumentation frameworks. *Argument & Computation* 1(2):147–177.

Ellmauthaler, S., and Strass, H. 2014. The DIAMOND system for computing with abstract dialectical frameworks. In *Proc. COMMA*, volume 266 of *FAIA*, 233–240.

Gaggl, S. A.; Manthey, N.; Ronca, A.; Wallner, J. P.; and Woltran, S. 2015. Improved answer-set programming encodings for abstract argumentation. *TPLP* 15(4-5):434–448.

Gebser, M.; Kaminski, R.; Kaufmann, B.; and Schaub, T. 2012. *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers.

Gogic, G.; Kautz, H.; Papadimitriou, C.; and Selman, B. 1995. The comparative linguistics of knowledge representation. In *Proc. IJCAI*, 862–869.

Gomes, C. P.; Kautz, H. A.; Sabharwal, A.; and Selman, B. 2008. Satisfiability Solvers. In *Handbook of Knowledge Representation*, volume 3 of *Foundations of AI*. Elsevier. 89–134.

Linsbichler, T.; Spanring, C.; and Woltran, S. 2015. The hidden power of abstract argumentation semantics. In *Proc. TAFA*, volume 9524 of *LNCS*, 146–162.

Marek, V. W., and Truszczyński, M. 1999. Stable models and an alternative logic programming paradigm. In *In The Logic Programming Paradigm: a 25-Year Perspective*. Springer. 375–398.

Modgil, S. 2009. Reasoning about preferences in argumentation frameworks. *Artif. Intell.* 173(9–10):901–934.

Nielsen, S. H., and Parsons, S. 2006. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Proc. ArgMAS*, volume 4766 of *LNCS*, 54–73.

Niemelä, I. 1999. Logic programs with stable model semantics as a constraint programming paradigm. *Ann. Math. Artif. Intell.* 25(3-4):241–273.

Polberg, S. 2016. *Developing and Extending the Abstract Dialectical Framework*. Ph.D. Dissertation, TU Wien, Austria. Upcoming.

Pührer, J. 2015. Realizability of Three-Valued Semantics for Abstract Dialectical Frameworks. In *Proc. IJCAI*, 3171–3177.

Strass, H., and Wallner, J. P. 2015. Analyzing the Computational Complexity of Abstract Dialectical Frameworks via Approximation Fixpoint Theory. *Artif. Intell.* 226:34–74.

Strass, H. 2015. Expressiveness of Two-Valued Semantics for Abstract Dialectical Frameworks. *J. Artif. Intell. Res. (JAIR)* 54:193–231.

Using Enthymemes to Fill the Gap between Logical Argumentation and Revision of Abstract Argumentation Frameworks

Jean-Guy Mailly
Institute of Information Systems
TU Wien, Autria
jmailly@dbai.tuwien.ac.at

Abstract

In this paper, we present a preliminary work on an approach to fill the gap between logic-based argumentation and the numerous approaches to tackle the dynamics of abstract argumentation frameworks. Our idea is that, even when arguments and attacks are defined by means of a logical belief base, there may be some uncertainty about how accurate is the content of an argument, and so the presence (or absence) of attacks concerning it. We use enthymemes to illustrate this notion of uncertainty of arguments and attacks. Indeed, as argued in the literature, real arguments are often enthymemes instead of completely specified deductive arguments. This means that some parts of the pair (support, claim) may be missing because they are supposed to belong to some "common knowledge", and then should be deduced by the agent which receives the enthymeme. But the perception that agents have of the common knowledge may be wrong, and then a first agent may state an enthymeme that her opponent is not able to decode in an accurate way. It is likely that the decoding of the enthymeme by the agent leads to mistaken attacks between this new argument and the existing ones. In this case, the agent can receive some information about attacks or arguments acceptance statuses which disagree with her argumentation framework. We exemplify a way to incorporate this new piece of information by means of existing works on the dynamics of abstract argumentation frameworks.

Introduction

Argumentation frameworks (AFs) are a convenient way to represent conflicting information and to deduce which subset of the information can be inferred. For instance, they can be used to model dialogs between several agents (Amgoud and Hameurlain 2006) or to analyze on-line discussion between social network users (Leite and Martins 2011). Argumentation can also be useful in a mono-agent setting, for instance to infer non-trivial conclusions from an inconsistent knowledge base (Besnard and Hunter 2001).

The domain called *dynamics of argumentation* has become a hot topic in recent years, with numerous publications about it. The first ones consider really classical debate scenarios as the source of the dynamic process (Boella, Kaci, and van der Torre 2009a; 2009b; Cayrol, de Saint-Cyr, and Lagasquie-Schiex 2010; Baumann and Brewka 2010; Bisquert et al. 2011; 2013; Baumann 2012; Booth et al. 2013). These approaches are perfectly well-suited for classical exchange of arguments between agents. Then, some approaches have proposed to consider new scenarios, closer to what happens with belief change in logical settings (Alchourrón, Gärdenfors, and Makinson 1985; Katsuno and Mendelzon 1991; 1992): these approaches propose to question the existing relation between arguments, and to modify this relation if it is required (Doutre, Herzig, and Perrussel 2014; Nouioua and Würbel 2014; Coste-Marquis et al. 2014a; 2014b; 2015; Baumann and Brewka 2015; Diller et al. 2015).

These works directly deal with the structure of the abstract AFs. An interesting question is "What does AF revision mean when we consider logic-based AFs?". Indeed, it is not obvious that attacks between arguments can be changed, since they stem from the logical inference relation; for instance, if arguments a and b attack each other because their claims are the negation of each other (*rebuttal* attack), then it is not accurate to consider that the attack between a and b could be removed. But this is only the case when we consider completely specified deductive arguments (Besnard and Hunter 2001). As argued in the literature (Hunter 2007), the arguments which are used in real situations are often enthymemes, which are partially specified arguments: some parts of the support or some parts of the claim are not described, because it is supposed that they belong to some "common knowledge". There may be different reasons for an agent not to share some part of her knowledge, such as some cost on the communication process. Then, the agent who receives an enthymeme must decide how to complete the content of the enthymeme to be able to use it. But if the missing formulae to complete the enthymeme are not part of the agent's beliefs (or at least, are not considered by the agent as the most accurate way to complete the enthymeme), then she will use a badly completed enthymeme in her argumentation framework. We see with this situation that, even with an underlying logical belief base, the nature of arguments and attacks is not absolute; it depends on the agent's beliefs and on her way to complete enthymemes.

So we propose to consider the use of enthymemes in the argumentation process to explain the questionability of some attacks. We illustrate the possibility that a logic-based argumentation framework contains mistaken attacks. Then we show that the existing work on the dynamics of AFs can be used on such enthymeme-based AFs, as soon as a distinction between classical deductive arguments and enthymemes is done in the abstract AF, and that this distinction is used in the revision process.

The paper is organized as follows. The first section presents the background notions required to the understanding of the paper. In particular, we describe briefly belief revision, abstract argumentation and revision of AFs, and logic-based argumentation. Then in the second section, we focus on enthymemes in logic-based AFs; we explain how using enthymemes can be a source of mistaken attacks in the resulting AF. The following section illustrates the revision process on logic-based AFs which contain enthymemes. After the description of a basic approach in which each attack concerning an enthymeme is questionable, we propose a refinement of this approach based on the notion of fixed part of an enthymeme. Finally, the last section concludes the paper and sketches some interesting future work.

Background

Belief Revision

Belief revision is well-known when an agent's beliefs are represented in a logical setting. The intuitive idea is "How can an agent incorporate a new piece of information into her beliefs?", which is not a trivial question when the agent's previous beliefs and the new piece of information are conflicting. One of the most influencial works on this topic is the AGM framework (Alchourrón, Gärdenfors, and Makinson 1985), which gives rationality postulates for belief change operators, when the beliefs are represented as deductively closed sets of formulae. Here we are interested in the adaptation of AGM revision to finite propositional logic by Katsuno and Mendelzon (1991). They explain that revising a formula φ by a formula α is equivalent to selecting some models of α which are minimal w.r.t. some plausibility relation. This relation has to satisfy some properties.

Definition 1 (Katsuno and Mendelzon 1991)**.** A faithful assignment is a mapping from a formula φ to a total pre-order between interpretations $\leq_\varphi$ such that:

1. if $I \models \varphi$ and $I' \models \varphi$, then $I \simeq_\varphi I'$;
2. if $I \models \varphi$ and $I' \not\models \varphi$, then $I <_\varphi I'$;
3. if $\varphi \equiv \varphi'$, then $\leq_\varphi = \leq_{\varphi'}$.

Then, a KM revision operator $\circ$ is a mapping from two formulae φ, α to a new formula such that

$$\text{mod}\ (\varphi \circ \alpha) = \min(\text{mod}\ (\alpha), \leq_\varphi)$$

For instance, the Dalal revision operator can be defined through the pre-order built on the Hamming distance.

Definition 2 (Hamming 1950; Dalal 1988)**.** The *Hamming distance* between two propositional interpretations I, I' is the number of assignments which differ between I and I', formally: $d_H(I, I') = |(I \backslash I') \cup (I' \backslash I)|$.
The total pre-order $\leq_\varphi^{d_H}$ is defined by

$$I \leq_\varphi^{d_H} I' \text{ iff } \min_{J \in \text{ mod}\,(\varphi)} (d_H(I, J)) \leq \min_{J \in \text{ mod}\,(\varphi)} (d_H(I', J))$$

The Dalal revision operator $\circ_D$ is a mapping from two formulae φ, α to a new formula such that

$$\text{mod}\ (\varphi \circ_D \alpha) = \min(\text{mod}\ (\alpha), \leq_\varphi^{d_H})$$

Let us illustrate the behavior of the Dalal revision operator.

Example 1. Consider $V = \{a, b, c, d\}$ and $\varphi = [(a \wedge b) \vee (\neg a \wedge c) \vee \neg(b \vee (a \wedge c))] \wedge \neg d$. The models of φ are $\{\{a\}, \{c\}, \{a, b\}, \{b, c\}, \{a, b, c\}\}$. We revise φ by $\alpha = a \wedge \neg b \wedge c$. The models of α are $\{\{a, c\}, \{a, c, d\}\}$. Table 1 gives the Hamming distance between models of φ and models of α.

	$\{a, c\}$	$\{a, c, d\}$
$\emptyset$	2	3
$\{a\}$	1	2
$\{c\}$	1	2
$\{a, b\}$	2	3
$\{b, c\}$	2	3
$\{a, b, c\}$	1	2

Table 1: Hamming distance between models of φ and α

Since the minimal Hamming distance between $\{a, c\}$ and a model of φ is 1 ($d_H(\{a, c\}, \{a\})$ for instance), while the distance between $\{a, c, d\}$ and any model of φ is at least 2, then $\{a, c\} <_\varphi^{d_H} \{a, c, d\}$, and so $\text{mod}\ (\varphi \circ_{Da} \alpha) = \{\{a, c\}\}$.

Abstract Argumentation and AF Revision

An abstract AF is a directed graph which represents the arguments and the attacks between them. The usual problem to solve with such an abstract AF is "How to determine which arguments are accepted?". This question is tackled in the seminal paper by Dung (1995).

Definition 3 (Dung 1995)**.** An argumentation framework (AF) is a pair $F = \langle A, R \rangle$ where A is a set of abstract entities called *arguments*, and $R \subseteq A \times A$ is the *attack relation* which represents the conflicts between arguments. Given a *semantics* σ, the σ-extensions of F, denoted $\sigma(F)$, are subsets of A which can be accepted. An argument is then *skeptically accepted* by F w.r.t. σ iff it belongs to each σ-extension of F.

In this paper, we illustrate our approach on the *stable semantics*: $S \subseteq A$ is a *stable extension* of F (denoted by $S \in st(F)$) iff

- $\nexists x, y \in S$ s.t. $(x, y) \in R$;
- $\forall y \in A \backslash S, \exists x \in S$ s.t. $(x, y) \in R$.

Example 2. Given the set of arguments $A = \{x, y, z, t, u\}$, the AF $F_1 = \langle A, R\rangle$ with $R = \{(x, y), (x, t), (y, x), (y, z), (z, u), (t, u)\}$ is given in Figure 1.

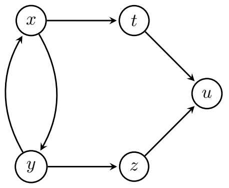


Figure 1: The AF F_1

Its stable extensions are $st(F_1) = \{\{x, z\}, \{y, t\}\}$.

As explained in the introduction, the question of change in AFs has been tackled by several approaches. Here we use the *translation-based revision* from (Coste-Marquis et al. 2014b). The idea of this method is to translate the AF and the semantics into a propositional formula, to use a KM revision operator to perform the expected change, and then to decode the models of the revised formula to obtain a set of revised AFs. The propositional encoding is a generalization of a result from Besnard and Doutre (2004). They have defined a formula Ξ, built on propositional variables corresponding to the arguments, such that the set of models of Ξ exactly correspond to the set of stable extensions of an AF. Coste-Marquis et al. (2014b) generalize this encoding with the addition of two other kinds of variables $V = \{att_{x,y} \mid x, y \in A\} \cup \{acc_x \mid x \in A\}$. $att_{x,y}$ means that there is an attack from the argument x to the argument y, and acc_x means that the argument x is skeptically accepted.

Definition 4 (Coste-Marquis et al. 2014b)**.** Given an AF $F = \langle A = \{x_1, \ldots, x_n\}, R\rangle$, the stable encoding of F is

$$f_{st}(F) = (\bigwedge_{(x,y)\in R} att_{x,y}) \wedge (\bigwedge_{(x,y)\notin R} \neg att_{x,y}) \wedge th_{st}(A)$$

where

$$\begin{aligned} th_{st}(A) = \quad & \textstyle\bigwedge_{x\in A}[acc_x \Leftrightarrow \forall x_1, \ldots, \forall x_n, \\ & (\textstyle\bigwedge_{y\in A}(y \Leftrightarrow \bigwedge_{z\in A}(att_{z,y} \Rightarrow \neg z)) \Rightarrow x)] \end{aligned}$$

In general, $f_\sigma(F)$ can be defined for any semantics σ as soon as the formula Ξ exists; for semantics with a complexity higher than NP, we can consider for instance QBF encodings to define Ξ.

Then, the revision operator is defined as follow:

Definition 5 (Coste-Marquis et al. 2014b)**.** Given $\circ$ a KM revision operator and φ a propositional formula built from the set of variables V, the *translation-based revision operator* $\star_\circ$ is defined as

$$F \star_\circ \varphi = dec(f_\sigma(F) \circ (\varphi \wedge th_\sigma(A)))$$

with dec a mapping from a formula ψ to a set of AFs $\mathcal{F}$ such that each AF $F' \in \mathcal{F}$ corresponds to one of the models ω of ψ: (x, y) appears in F' iff $att_{x,y}$ is true in ω.

This general definition allows to change any attack and argument status as long as it is compatible with σ. If additional constraints should be satisfied,[1] the use of a constrained version is possible:

Definition 6 (Coste-Marquis et al. 2014b)**.** Given $\circ$ a KM revision operator and φ, μ two propositional formulae built from the set of variables V, the *constrained translation-based revision operator* $\star_\circ^\mu$ is defined as

$$F \star_\circ^\mu \varphi = dec(f_\sigma(F) \circ (\varphi \wedge th_\sigma(A) \wedge \mu))$$

To conclude, let us mention a particular revision operator proposed by (Coste-Marquis et al. 2014b): we call $\star_{att}$ (resp. $\star_{att}^\mu$) the translation-based (resp. constrained translation-based) revision operator which gives priority to the minimal change of the attack relation. This operator is similar to the Dalal-based revision revision operator $\star_{\circ_D}$, but it uses a weighted version of the Hamming distance such that changing the value of a single $att_{x,y}$ variable is more expensive than changing the value of each acc_x variable.

Example 3. We consider the AF F_1 given in Figure 1. We suppose the existence of an integrity constraint $att_{t,u} \wedge att_{z,u}$, which means that the attacks from t and z to u must not be removed. The result of the revision $F_1 \star_{\circ_D}^\mu acc_u$ is the AF F_2 described in Figure 2. Now the extensions are $st(F_2) = \{\{x, u\}, \{y, u\}\}$, so u is skeptically accepted.

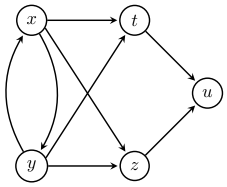


Figure 2: $F_2 = F \star_{\circ_D}^\mu acc_u$

We focus on this kind of revision operators because (Coste-Marquis et al. 2014b) already proposes a way to incorporate a constraint on the attack relation, which is required by our approach. Other revision or update operators could be used instead, but we should adapt their definition to take into account the constraint.

Logic-based Arguments: Deductive Arguments

The question of the exact nature of arguments and attacks is tackled by several approaches which can be gathered under the name *structural argumentation*. Here we focus on one of the most prominent ones: deductive argumentation (Besnard and Hunter 2001).

Definition 7 (Besnard and Hunter 2001)**.** A deductive argument built from a belief base Δ is a pair $\langle\Phi, \alpha\rangle$, where Φ is called the *support* and α the *claim*, such that:

1. $\Phi \subseteq \Delta$,
2. $\Phi \nvdash \bot$,
3. $\Phi \vdash \alpha$,
4. Φ is minimal with respect to $\subseteq$ among the sets of formulae which satisfy items 1. 2. and 3.

[1] Such as external constraint depending on the particular application, or some rules of the world.

There is an intuitive explanation to this definition. First the agent is supposed to use her beliefs to justify her claim, which explains the first condition. The second and third conditions guarantee that the claim is actually supported by the beliefs of the agent, but not by conflicting beliefs (for instance, the sentence "It is raining and it is not raining, so I am the Queen of England." is not an argument at all). Finally, the last condition ensures that there is no useless piece of information in the support: "It is raining, when it is raining I should use an umbrella, and I love chocolate. So I will use my umbrella." is not accurate either.

The conflicts between deductive arguments may have different natures. The most general sort of conflict is defined as follow:

Definition 8 (Besnard and Hunter 2001)**.** A *defeater* for an argument $\langle \Phi, \alpha \rangle$ is an argument $\langle \Phi', \alpha' \rangle$ such that $\alpha' \vdash \neg(\varphi_1 \wedge \cdots \wedge \varphi_n)$, for some $\{\varphi_1, \dots, \varphi_n\} \subseteq \Phi$.

It is possible to use deductive arguments to build an *argument tree* with the arguments and counterarguments which attack and defend a given claim.
We can also build a full argumentation framework from the set of deductive arguments generated from a belief base.

Definition 9 (Besnard and Hunter 2014)**.** Given A the set of deductive arguments generated from the belief base Δ, the *exhaustive graph* associated with Δ is the AF $F = \langle A, R \rangle$ with $R = \{(x, y) \in A \times A \mid x$ is a defeater for $y\}$.

Here we focus on the defeater relation, which is the most general one, but exhaustive graphs can be generated with another attack relation which guarantees additional properties for the defeaters (undercut, rebuttal, and so on). Moreover, these graphs may be infinite in general; Besnard and Hunter propose an approach to circumvent this problem. See (Besnard and Hunter 2014) for more details.

Enthymemes and their Role in Mistaken Attacks

Intuitive Explanation

Before formalizing our approach, we want to explain intuitively, with natural language arguments, why agents can disagree on the attack relation, and more generally why attacks could be questionable. Let us consider the following arguments:

(c) The US army is preparing a secret plan to retreat from Afghanistan (source: Wikileaks).

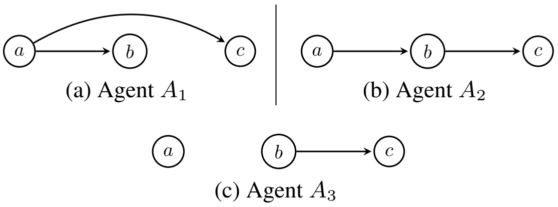
(b) Our informed sources say that the Wikileaks documents are fake (source: NY Times).

(a) The media cannot be trusted on military issues (source: N. Chomsky).

Now we consider three agents A_1, A_2, A_3; each of them may have some personal beliefs which are not shared with the other agents.

- A_1 thinks that Chomsky is the most credible source, and considers that Wikileaks is a media more reliable than NY Times. So her AF is the one given in Fig. 3a.
- A_2 thinks that Chomsky is a more credible source than NY Times, and NY Times is a more credible source than Wikileaks. She also believes that Wikileaks cannot be seen as a media. So her AF is the one given in Fig. 3b.
- Finally, A_3 thinks that NY Times is the most credible source, and that Chomsky is not reliable on this topic. So her AF is the one given in Fig. 3c.

These personal AFs may depend on many different parameters (additional information which is not available to each agent, preferences, context, previous experience of each agent, and so on).



(a) Agent A_1 (b) Agent A_2

(c) Agent A_3

Figure 3: Three Agents Disagreement

Of course, under the assumption that the agents share all their knowledge and beliefs, the personal beliefs of the agents can be represented as additional arguments and we obtain a single AF representing the whole information about a topic. But we think that this assumption is too strong for at least three reasons. First, there may be technical issues with this information sharing; for instance, there may be some cost on communication between agents, or the global amount of information in the network may be too important to be stored in a centralized way. Then, for strategical reasons, agents may *choose* not to share their knowledge and beliefs. Also, if argument are mined from natural language (for instance, for an analysis of social networks debates), there are likely some implicit pieces of information used in the argumentation process. This explains why some attacks may be questionable.

For instance, if the agents A_1 and A_3 consider that agent A_2 is trustworthy, then they could have to change the attack relation in their own AFs if they receive from agent A_2 the information "c should be accepted". On the opposite, if agents vote to determine the arguments statuses, there will be a majority of agents (A_1 and A_3) voting against c (meaning that c is rejected in their AFs), so agent A_2 should modify the attack relation to incorporate this piece of information in her AF.

Enthymemes with partial support Now let us formalize this notion of "arguments with partial knowledge", and their role in the existence of mistaken attacks. Hunter (2007) defines what he calls *approximate arguments*, which are pairs $\langle \Phi, \alpha \rangle$ which do *not* satisfy the four conditions of deductive arguments. He classifies them depending on which properties they satisfy, and then focuses on enthymemes. An *enthymeme* is a pair $\langle \Phi, \alpha \rangle$ such that $\Phi \not\vdash \alpha$, but there is a set $\Psi \subseteq \Delta$ such that $\langle \Phi \cup \Psi, \alpha \rangle$ is a deductive argument. Intuitively, Ψ represents some "common knowledge" that the agent supposes to be known by her opponents. Then it is not useful for the agent to state the full deductive argument to be able to exchange information and to reach her goal (persuading her opponent, helping to take a decision, negotiating, and so on).

Example 4. To illustrate this concept, we borrow a simple example of real life use of enthymemes from (Hunter 2007). Let us consider John and his wife Yoko, who is going outside without an umbrella. If John tells her "You should take your umbrella, because the weather report predicts rain", there is no formal reason to consider that $\langle \Phi, \alpha \rangle$ (with $\Phi = \{rain\_predicted\}$ and $\alpha = take\_umbrella$) is an argument. It is in fact an enthymeme, because John supposes that $\Psi = \{rain\_predicted \Rightarrow take\_umbrella\}$ is part of the knowledge he shares with Yoko.

One of the questions tackled in (Hunter 2007) is "How does the agent knows that Ψ is *actually* part of the common knowledge?". Hunter supposes that each agent has a way to evaluate the plausibility that a given formula will be part of the knowledge shared between her and another agent.

Definition 10 (Hunter 2007)**.** For each agent A_i whose beliefs are expressed in the propositional language $\mathcal{L}$,

- $\Delta_i \subseteq \mathcal{L}$ denotes her own *personal base*,
- and for each other agent A_j, $\mu_{i,j}$ is a mapping from the language $\mathcal{L}$ to $[0, 1]$, such that $\mu_{i,j}(\alpha)$ represents the certainty that α is common to both agents A_i and A_j.

On enthymemes and mistaken attacks This mapping $\mu_{i,j}$ is used by the agent to build her arguments and decide whether they should be fully specified deductive arguments, or whether enthymemes can be used. The idea is simply to keep only the formulae φ in the support such that the associated value $\mu_{i,j}(\varphi)$ is less than a given threshold τ; the other ones can be omited because they are supposed to be known by agent A_j.

This process may lead to some problems in the exchange of arguments. There are at least two sources of mistakes.

1. The mapping $\mu_{i,j}$ describes the *perception* that agent A_i has of her common knowledge with A_j. If this perception is wrong, then there could be some exchange of enthymemes that the agent A_j cannot decode accurately.
2. Even with a good evaluation of the common knowledge by $\mu_{i,j}$, the choice of a bad threshold could also lead to enthymemes that the other agent cannot decode.

In both these situations, agent A_j receives some "argument" $a = \langle \Phi, \alpha \rangle$ which is not fully specified, and then she has to complete the support with some Ψ' from her own belief base, which could of course lead to the addition of some attacks from an existing argument b to this new argument a, for instance if the claim of b is the formula $\neg\psi$, for some $\psi \in \Psi'$. Even if for low-level treatments, it can be represented as $a' = \langle \Phi \cup \Psi', \alpha \rangle$ (for instance to determine if possibly new incoming arguments attack it), at a higher level it is still the argument a which is used. Indeed, this a' is not an argument that A_j has built by herself, since some of the premises are not part of her belief base.[2] Then, agent A_j can receive a new piece of information about the argument a which is incompatible with the attack from b to a (the simplest example being "a and b should be accepted together"). So she has to build a new internal state a'' from a subset Ψ'' from her belief base; for the same reason as previously, at a higher level it is still the argument a originally built from agent A_i's beliefs.

Enthymemes with partial claim We have seen that enthymemes are a way to communicate arguments with partial support. Black and Hunter (2012) also give some examples of enthymemes with a partial claim. Borrowing their example, let us consider the sentence α = "John has bought The Times". The enthymeme $\langle \{\alpha\}, \top \rangle$ can be interpreted in at least two ways, which lead to different claims:

1. $\langle \{\alpha, \alpha \Rightarrow \beta\}, \beta \rangle$ with β = "John has bought a copy of the newspaper The Times";
2. $\langle \{\alpha, \alpha \Rightarrow \gamma\}, \gamma \rangle$ with γ = "John has bought the company which publishes the newspaper The Times".

Similarly to what we have described for enthymemes with a partial support, if an agent receives an argument a which is in fact an enthymeme with a partial claim, some mistakes in the attack relation can appear. For instance, she may consider that a attacks an argument b because some part of b's support is conflicting with the completed claim (either β or γ in our example). If she later receives a piece of information which is not compatible with this attack, then she may have to consider a removal of this attack (for instance, because the chosen claim is not accurate).

So we can formally define the class of enthymemes (with partial claims and partial supports) as follows:

Definition 11 (Black and Hunter 2012)**.** Given $d = \langle \Phi, \alpha \rangle$ a deductive argument, an approximate argument $\langle \Phi', \alpha' \rangle$ is an enthymeme for d iff $\Phi' \subset \Phi$ and $\alpha \vdash \alpha'$.

Stated otherwise, the pair $\langle \Phi, \alpha \rangle$ is an enthymeme for $\langle \Phi \cup \Psi, \alpha \wedge \beta \rangle$, with Φ the partial support and α the partial claim. In the rest of this paper, we call such a pair $\langle \Phi, \alpha \rangle$ a *non-completed enthymeme* and $\langle \Phi \cup \Psi, \alpha \wedge \beta \rangle$ a *completed enthymeme*. A completed enthymeme may not satisfy the conditions stated in Definition 7, since the set of formulae Φ comes from another agent's belief base. Moreover, contrary to a fully specified argument stemming from the agent's beliefs, a completed enthymeme can be questioned.

[2] To do this, it is a logical belief revision/expansion/update which should be performed, and this would likely have some side effects on the whole belief base, not only on the formulae involed in argument a.

Dynamics of AFs and Enthymemes

Building a Dung's AF from Enthymemes

For several reasons, the use of an abstract AF by the agent is interesting, even when she uses an underlying belief base. For instance, the developement of efficient approaches to solve abstract argumentation problems permits to obtain the conclusion of the agent's AF with respect to several semantics and inference policies (see for instance the competition of argumentation solvers (Thimm and Villata 2015)). But to avoid the loss of information about the nature of arguments and attacks, we propose to refine the definition of the AF.

Definition 12. Given D and E which denote respectively the agent's deductive arguments and enthymemes, the agent's *enthymeme-based AF* is $F(D, E) = \langle A, R \rangle$ with

- $A = D \cup E$;
- $R = R_D \cup R_E$;
- $R_D \subseteq D \times D$ the set of certain attacks (between deductive arguments);
- $R_E \subseteq (A \times A) \backslash (D \times D)$ the set of questionable attacks (concerning at least one enthymeme).

Computing the extensions of such an AF is identical to the process for classical Dung's AFs; differentiating both kinds of attacks is useful only for the dynamics scenarios such as revision.

In this setting, each attack can be added or removed as soon as it concerns at least one enthymeme. We will refine this later.

Example 5. Let F_3 be the enthymeme-based AF presented in Fig. 4. Arguments with rounded corners are the enthymemes while the other ones are deductive arguments. Similarly, the dashed arrows represent the questionable attacks, while the other ones are the certain attacks. In this example, we suppose that the agent has received the enthymemes in the following way:

- $e_1 = \langle \{\alpha\}, \gamma \rangle$, which has been completed by $\Psi_1 = \{\alpha \Rightarrow \beta, \beta \Rightarrow \gamma\}$;
- $e_2 = \langle \{\eta\}, \top \rangle$, which has been completed by $\Psi_2 = \{\eta \Rightarrow \neg\epsilon\}$ in the support and $\neg\epsilon$ in the claim.

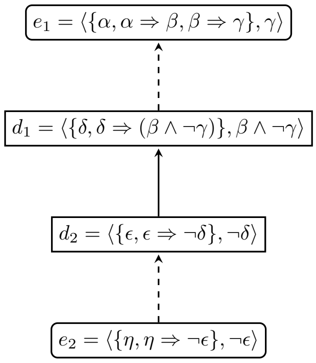

$e_1 = \langle \{\alpha, \alpha \Rightarrow \beta, \beta \Rightarrow \gamma\}, \gamma \rangle$
$d_1 = \langle \{\delta, \delta \Rightarrow (\beta \wedge \neg\gamma)\}, \beta \wedge \neg\gamma \rangle$
$d_2 = \langle \{\epsilon, \epsilon \Rightarrow \neg\delta\}, \neg\delta \rangle$
$e_2 = \langle \{\eta, \eta \Rightarrow \neg\epsilon\}, \neg\epsilon \rangle$

Figure 4: The Enthymeme-based AF F_3

With this AF, the accepted arguments are $\{e_2, d_1\}$.

Applying Dynamics of Abstract AFs to Enthymeme-based AFs

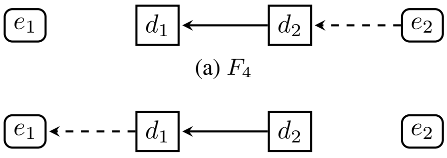
The existence of mistaken attacks in an enthymeme-based AF can be tackled through some approaches of the *dynamics of abstract argumentation* (Bisquert et al. 2013; Doutre, Herzig, and Perrussel 2014; Coste-Marquis et al. 2014a; 2014b; 2015). In the case when some arguments and the relations between them are certain (in particular, when they are fully specified arguments instead of enthymemes), integrity constraints can simply be added to these revision/update/enforcement operators to ensure that forbidden attacks will not be added, and mandatory attacks will not be removed. Since it is already defined by (Coste-Marquis et al. 2014b), we will exemplify the dynamics of argumentation with their *constrained revision* approach, presented previously. We can encode an integrity constraint to fix the attacks and non-attacks concerning the deductive arguments into the setting from (Coste-Marquis et al. 2014b).

Definition 13. Given $F(D, E)$ an enthymeme-based AF, the integrity constraint on deductive arguments is

$$\mu_D = \Big(\bigwedge_{(x,y) \in R_D} att_{x,y} \Big) \wedge \Big(\bigwedge_{(x,y) \in (D \times D) \backslash R_D} \neg att_{x,y} \Big)$$

Now, if the agent receives some piece of information about the arguments statuses or the attack relation, then she can use the AF revision operator $\star_{att}^{\mu_D}$ as defined previously in the case when this new piece of information disagrees with the current AF. This revision operator guarantees that the relations between deductive arguments will not be modified during the revision process, which is desirable since they are directly stemming from the logical inference relation.

Example 5 Continued. We continue the previous example. The agent receives the piece of information "e_1 should be accepted", which corresponds to the formula acc_{e_1}. The integrity constraint is $att_{d_2,d_1} \wedge \neg att_{d_1,d_2}$, which ensures that the attacks between the deductive arguments d_1 and d_2 will not be modified. The possible results are given in Fig. 5.



(a) F_4

(b) F_5

Figure 5: Possible Results of the Revision

The exact *change operator* which should be used depends on the the properties expected for the process, for instance it is well-known that performing an update (Bisquert et al. 2013; Doutre, Herzig, and Perrussel 2014) is accurate when the change is explained by an evolution of the world, while performing a revision (Coste-Marquis et al. 2014a; 2014b) is accurate when the evolution only concerns the agent's beliefs about the world; thus these operations do not satisfy the same properties. Similarly, among the different approaches in the state of the art about the dynamics of AFs, each of them do not have the same expressivity. For instance, the revision approach described in (Coste-Marquis et al. 2014a) permits to revise by a formula concerning the extensions, while the translation-based approach illustrated here permits to revise by a formula concerning skeptical acceptance of arguments and attacks at the same time. So, the choice of a change operator completely depends on the application and the agent's needs and preferences. In the following, we continue to consider revision to be consistent with the previous example, but update and extension enforcement (Baumann and Brewka 2010) could be considered as well.

From Revised AFs to new Completed Enthymemes

After obtaining the result of the revision process, the agent should now decode this result to determine which of the revised AFs is the most plausible real AF corresponding to her beliefs, and which enthymemes should be internally modified (and *how* they should be internally modified) to ensure that the abstract AF and the logic-based AF coincide.

Definition 14. Let $\mathcal{F}$ be the set of AFs obtained from the revision process. For each $F' \in \mathcal{F}$, F' is called an *acceptable AF* iff for each attack which differs between the original AF F and F', the agent's belief base contains some formulae which allow to complete the enthymemes s.t. this new completion is consistent with the attacks in F'.

Example 5 Continued. Continuing the previous example, let us suppose that the agent's belief base contains the formulae $\Psi' = \{\alpha \Rightarrow \theta, \theta \Rightarrow \gamma\}$. Then the enthymeme e_1 can be completed into $\langle\{\alpha, \alpha \Rightarrow \theta, \theta \Rightarrow \gamma\}, \gamma\rangle$, which leads to the acceptable AF F_4 given in Fig. 6a. Similarly, if the agent's belief base contains the formulae $\Psi'' = \{\eta \Rightarrow \iota\}$, then the agent can consider the acceptable AF F_5 given in Fig. 6b, since e_2 can be completed into $\langle\{\eta, \eta \Rightarrow \iota\}, \iota\rangle$.

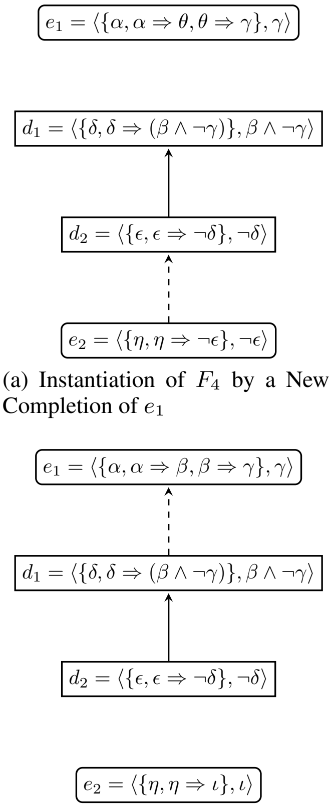
When the set of acceptable AFs is not a singleton, we can consider two different solutions:

- the agent can keep the whole set as the result, to express the uncertainy of the result of the revision, as suggested by (Bisquert et al. 2013; Coste-Marquis et al. 2014a; 2014b; Doutre, Herzig, and Perrussel 2014) which consider that revising or updating an AF can lead to a set of AFs;
- the agent can use external information (preferences between AFs, preferences between formulae in the enthymemes, and so on) to select a single acceptable AF as the result.

None of them is in general more desirable than the other one, the choice depends on the situation (specific application, user's preferences, computational issues,. . .).

Refining Questionable Attacks

In the previous parts, we suppose that each attack concerning an enthymeme is questionable. But we can be more precise in the definition of the enthymeme-based AF. Indeed, even when we consider an enthymeme e, some of the attacks concerning it may be certain. We know that some parts of e, that we have called previously the partial support and the partial claim, are fixed. If the reason of an attack between e and a deductive argument is a logical conflict involving one of these fixed parts of e, then this attack can be considered as certain. Similarly, when we consider another enthymeme e', if there is an attack between e and e' which is stemming from the fixed part of e and the fixed part of e', then this attack cannot be removed either.


$e_1 = \langle\{\alpha, \alpha \Rightarrow \theta, \theta \Rightarrow \gamma\}, \gamma\rangle$
$d_1 = \langle\{\delta, \delta \Rightarrow (\beta \wedge \neg\gamma)\}, \beta \wedge \neg\gamma\rangle$
$d_2 = \langle\{\epsilon, \epsilon \Rightarrow \neg\delta\}, \neg\delta\rangle$
$e_2 = \langle\{\eta, \eta \Rightarrow \neg\epsilon\}, \neg\epsilon\rangle$

(a) Instantiation of F_4 by a New Completion of e_1

$e_1 = \langle\{\alpha, \alpha \Rightarrow \beta, \beta \Rightarrow \gamma\}, \gamma\rangle$
$d_1 = \langle\{\delta, \delta \Rightarrow (\beta \wedge \neg\gamma)\}, \beta \wedge \neg\gamma\rangle$
$d_2 = \langle\{\epsilon, \epsilon \Rightarrow \neg\delta\}, \neg\delta\rangle$
$e_2 = \langle\{\eta, \eta \Rightarrow \iota\}, \iota\rangle$

(b) Instantiation of F_5 by a New Completion of e_2

Figure 6: Different Acceptable AFs

Let us first formalize this notion of fixed part.

Definition 15. If $a = \langle\Phi, \alpha\rangle$ is a deductive argument or a non-completed enthymeme, then the *fixed part* of a is $fix(a) = \Phi \cup \{\alpha\}$.
If $a = \langle\Phi \cup \Psi, \alpha \wedge \beta\rangle$ is a completed enthymeme, then $fix(a) = \Phi \cup \{\alpha\}$.

So if we consider a fully specified deductive argument or a non-completed enthymeme, the fixed part is the set of all the formulae involved in it. But when we consider an enthymeme *completed with the agent's beliefs*, then the fixed part is the set of formulae which appear in the enthymeme that the agent has originally received, but do not appear in the completed version of it.

Example 5 Continued. Let us consider again the arguments d_1, d_2, e_1 and e_2. The fixed parts of the deductive arguments

are trivially the union of their support and their claim.
The result is more interesting for the enthymemes:

- $fix(e_1) = \{\alpha, \gamma\}$;
- $fix(e_2) = \{\eta, \top\}$;

Now let us define the involved part of an argument in an attack.

Definition 16. Let $a = \langle \Phi, \alpha \rangle$ and $b = \langle \Phi', \alpha' \rangle$ be two arguments (deductive arguments, completed enthymemes or non-completed enthymemes). If there is an attack between a and b, then the *involved part* of a in the conflict between a and b, denoted by $inv_b(a)$, is the set $\Psi \subseteq \Phi \cup \{\alpha\}$ such that $(\bigwedge_{\psi \in \Psi} \psi) \wedge (\bigwedge_{\varphi' \in \Phi' \cup \{\alpha'\}} \varphi') \vdash \bot$ and Ψ is minimal w.r.t. $\subseteq$. Otherwise, $inv_b(a) = inv_a(b) = \emptyset$.

So, if we have a rebuttal conflict between a and b (meaning that the claims are the contradiction of each other) then $inv_b(a) = \{\alpha\}$ and $inv_a(b) = \{\alpha'\}$. If the conflict is an undercut from a to b (meaning that the claim α of a is conflicting with some part φ' of the support of b), then $inv_b(a) = \{\alpha\}$ and $inv_a(b) = \{\varphi'\}$.

Example 5 Continued. Now we can see which parts of the arguments d_1, d_2, e_1 and e_2 are involved in conflicts. The certain attack (d_2, d_1) comes from the contradiction between δ and $\neg\delta$, so $inv_{d_1}(d_2) = \{\neg\delta\}$ and $inv_{d_2}(d_1) = \{\delta\}$. Concerning the questionable attacks, we have:

- $inv_{e_1}(d_1) = \{\beta \wedge \neg\gamma\}$ and $inv_{d_1}(e_1) = \{\beta \Rightarrow \gamma\}$;
- $inv_{e_2}(d_2) = \{\epsilon\}$ and $inv_{d_2}(e_2) = \{\neg\epsilon\}$.

Now we can refine the definition of an enthymeme-based AF.

Definition 17. Given D and E which denote respectively the agent's deductive arguments and enthymemes, the agent's *refined enthymeme-based AF* is $F(D, E) = \langle A, R \rangle$ with

- $A = D \cup E$;
- $R = R_C \cup R_Q$;
- $R_C = \{(x, y) \in A \times A \mid inv_y(x) \subseteq fix(x)$ and $inv_x(y) \subseteq fix(y)\}$: the set of certain attacks;
- $R_Q \subseteq (A \times A) \setminus R_C$: the set of questionable attacks.

We use R_D as a notation for $R_C \cap (D \times D)$, which is the set of attacks between deductive arguments.

Of course, if an argument is a fully specified deductive argument, then the part of it which is involved in conflicts is a fixed part. So to refine the AF, we need to check if it is the case with the enthymemes.

Example 5 Continued. Studying the relations between involved parts and fixed parts for the enthymemes e_1 and e_2, we obtain the following:

- $inv_{d_1}(e_1) = \{\beta \Rightarrow \gamma\} \not\subseteq fix(e_1) = \{\alpha, \gamma\}$;
- $inv_{d_2}(e_2) = \{\neg\epsilon\} \not\subseteq fix(e_2) = \{\eta, \top\}$.

So none of the attacks (e_2, d_2) and (d_1, e_1) is certain.

But we can exhibit more interesting cases, for which the use of a refined enthymeme-based AF leads to another result than the basic enthymeme-based AF.

Example 6. Let $d_3 = \langle \{\nu, \nu \Rightarrow \neg\lambda\}, \neg\lambda \rangle$ be a deductive argument, and $e_3 = \langle \{\kappa\}, \lambda \rangle$ an enthymeme, which can be completed for instance by the additional support $\Phi' = \{\kappa \Rightarrow \lambda\}$.
It is easy to see here that $inv_{d_3}(e_3) = \{\lambda\} \subseteq fix(e_3) = \{\kappa, \lambda\}$, so the conflict between d_3 and e_3 is not questionable, and the AF corresponding to these arguments is F_6 given in Fig. 7.

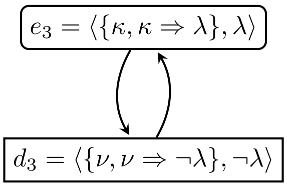

$e_3 = \langle \{\kappa, \kappa \Rightarrow \lambda\}, \lambda \rangle$
$d_3 = \langle \{\nu, \nu \Rightarrow \neg\lambda\}, \neg\lambda \rangle$

Figure 7: The Refined Enthymeme-based AF F_6

When we consider these refined enthymeme-based AFs in the revision process, the integrity constraint must be adapted to take into account each certain attack, and not only the ones between deductive arguments:

Definition 18. Given $F(D, E)$ a refined enthymeme-based AF, the integrity constraint on certain attacks is

$$\mu_C = (\bigwedge_{(x,y) \in R_C} att_{x,y}) \wedge (\bigwedge_{(x,y) \in (D \times D) \setminus R_D} \neg att_{x,y})$$

This new constraint ensures that a certain attack will not be removed during the revision process, and that attacks between deductive arguments will not be added if they do not belong to the original AF.

Back to Chomsky Example

To conclude, let us formalize the intuitive "Chomsky example", showing the different completions of enthymemes which lead to the different agents AFs. We use the following propositional variables: $retreat$ means that the US army will retreat; wkr means that the **Wik**ileaks information about **r**etreat is true; wkf means that the **Wik**ileaks documents are **f**ake; mnt means that **m**edia can **n**ot be **t**rusted on military issues. As they are stated, the arguments a, b and c which are shared by the agents are these ones:

$$a = \langle \{mnt\}, \top \rangle, b = \langle \{wkf\}, \top \rangle, c = \langle \{wkr\}, \top \rangle$$

All of them are enthymemes. For all the agents, the completion of c is $\langle \{wkr, wkr \Rightarrow retreat\}, retreat \rangle$. But they disagree on the completion of the other enthymemes. Agent A_1 considers that $a = \langle \{mnt, mnt \Rightarrow \neg wkf, mnt \Rightarrow \neg wkr\}, \neg wkf \wedge \neg wkr \rangle$ and $b = \langle \{wkf\}, \top \rangle$. Agent A_2 completes the enthymemes as follows: $a = \langle \{mnt, mnt \Rightarrow \neg wkf\}, \neg wkf \rangle$ and $b = \langle \{wkf, wkf \Rightarrow \neg wkr\}, \neg wkr \rangle$.

Finally, agent A_3 uses these completions of enthymemes: $a = \langle \{mnt\}, \top \rangle$ and $b = \langle \{wkf, wkf \Rightarrow \neg wkr\}, \neg wkr \rangle$.

These completions of enthymemes lead to the AFs described in Figure 3, with all arguments which are enthymemes, and all attacks which are questionable. So here, in case of a revision, the revision operator is used with the integrity constraint $\top$, which is equivalent to a revision without a constraint.

We mentioned in the introduction two scenarios which require to use dynamics of argumentation techniques. First, we suppose that agent A_2 is considered to be trustworthy by other agents. Then, when she says that c should be accepted (which is represented by the formula acc_c), the other agents have to revise their AF with this new piece of information. The result of the revision for A_1, with a corresponding completion of enthymemes which are modified because of the revision, is given in Figure 8.

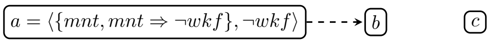

$a = \langle\{mnt, mnt \Rightarrow \neg wkf\}, \neg wkf\rangle$
b
c

Figure 8: Revision for Agent A_1

Similarly, for A_3, Figure 9 describes the possible revised AFs, with the modified enthymemes corresponding to it.

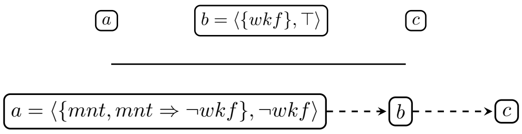

a
$b = \langle\{wkf\}, \top\rangle$
c
$a = \langle\{mnt, mnt \Rightarrow \neg wkf\}, \neg wkf\rangle$
b
c

Figure 9: Possible Revisions for Agent A_3

Finally, the other scenario was a vote on the acceptance status of c. Since the majority of agents rejects c, A_2 has to revise her AF by $\neg acc_c$ to find an agreement with the majority. Possible results are described in Figure 10.

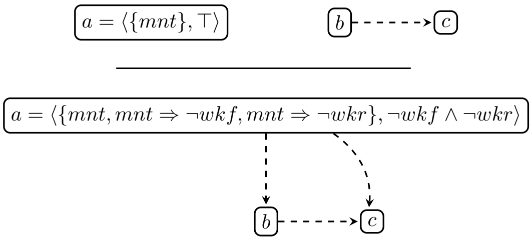

$a = \langle\{mnt\}, \top\rangle$
b
c
$a = \langle\{mnt, mnt \Rightarrow \neg wkf, mnt \Rightarrow \neg wkr\}, \neg wkf \wedge \neg wkr\rangle$
b
c

Figure 10: Possible Revisions for Agent A_2

Conclusion

In this paper, we argue that, in realistic situations, agents do not share *all* their knowledge and beliefs. There are different possible reasons, among them, technical and strategical reasons seem to be the most intuitive explanations. Also, implicit information is frequently used in natural language argumentation (on social networks for instance). When this situation occurs, there is some uncertainty in the resulting argumentation frameworks built by the agents. It is likely that agents' opinion about arguments' meaning and relations between arguments will differ; there may be some misunderstanding in the communication process which leads to mistakes in the generation of arguments and attacks. Here, this is formalized with the use of enthymemes, instead of deductive arguments, to represent the uncertain nature of some arguments. For this reason, the reception of a new piece of information (supposed to be reliable) can force the agents to question the current attack relation to obtain a result which is compatible with the new piece of information. We have described formally how the use of enthymemes in the argumentation process can lead to the existence of these mistaken attacks, and how to define an argumentation framework which makes the distinction between the certain attacks and the questionable attacks. Then, we have seen that the existing works on the dynamics of abstract AFs can be used to perform a change on an enthymeme-based AF when it is required to incorporate a new piece of information. Here, we exemplify it with the translation-based revision from (Coste-Marquis et al. 2014b), but it can be adapted to any revision, update or enforcement approach as soon as it is possible to consider an integrity constraint on the attack relation.

This paper only presents some preliminary work on this question. Many future works can be envisioned. First, we want to model the uncertainty by other means than enthymemes. For instance, using weights could lead to the definition of original change operators which define the notion of minimal change w.r.t. these weights; it would then be more expensive to change a single attack which has a high weight than to change several attacks with low weights. Determining what can be the origin of these weights is particularly interesting. Combined with the use of enthymemes, we think that giving a low weight to an attack which is easy to modify (because there are many possible completions of enthymemes in the belief base) is an interesting way to tackle the problem. For the EAFs defined in this paper, as well as the weighted approach mentioned above, several questions are opened. We have made the simplifying hypothesis that the agents will have some possible completion of arguments at their disposal, which is not always the case in real world situations. Similarly, the revision operators may lead to an empty-set of results (because of the integrity constraint), or on the opposite, to a non-singleton set of results. All these cases must be investigated to ensure the possibility of some practical applications. The complexity of revising such framework, compared to the original revision approach in the abstract setting, will also be studied to be able to identify which approaches can be used for real applications. We also plan to use some of the existing pieces of software for the dynamics of AFs (in particular the one described in (Coste-Marquis et al. 2015; Wallner, Niskanen, and Järvisalo 2015) for extension enforcement), to study the scalability of the approaches on practical examples. But it requires first an important work to build argumentation graphs from logical knowledge-bases, since the existing works focus only on argumentation trees (Efstathiou and Hunter 2008; 2011; Besnard et al. 2010).

Acknowledgments

This work as been supported by the Austrian Science Fund (FWF) under grants P25521 and I1102.

References

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change : Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50:510–530.

Amgoud, L., and Hameurlain, N. 2006. An argumentation-based approach for dialog move selection. In *Proc. of ArgMAS 2006*, 128–141.

Baumann, R., and Brewka, G. 2010. Expanding argumentation frameworks: Enforcing and monotonicity results. In *Proc. of COMMA 2010*, 75–86.

Baumann, R., and Brewka, G. 2015. AGM meets abstract argumentation: Expansion and revision for Dung frameworks. In *Proc. of IJCAI 2015*.

Baumann, R. 2012. What does it take to enforce an argument? minimal change in abstract argumentation. In *Proc. of ECAI 2012*, 127–132.

Besnard, P., and Doutre, S. 2004. Checking the acceptability of a set of arguments. In *Proc. of NMR 2004*, 59–64.

Besnard, P., and Hunter, A. 2001. A logic-based theory of deductive arguments. *Artificial Intelligence* 128(1-2):203–235.

Besnard, P., and Hunter, A. 2014. Constructing argument graphs with deductive arguments: A tutorial. *Argument and Computation* 5(1):5–30.

Besnard, P.; Grégoire, E.; Piette, C.; and Raddaoui, B. 2010. MUS-based generation of arguments and counter-arguments. In *Proc. of IRI 2010*, 239–244.

Bisquert, P.; Cayrol, C.; de Saint-Cyr, F. D.; and Lagasquie-Schiex, M.-C. 2011. Change in argumentation systems: Exploring the interest of removing an argument. In *Proc. of SUM 2011*, 275–288.

Bisquert, P.; Cayrol, C.; de Saint-Cyr, F. D.; and Lagasquie-Schiex, M. 2013. Enforcement in argumentation is a kind of update. In *Proc. of SUM 2013*, 30–43.

Black, E., and Hunter, A. 2012. A relevance-theoretic framework for constructing and deconstructing enthymemes. *Journal of Logic and Computation* 22(1):55–78.

Boella, G.; Kaci, S.; and van der Torre, L. 2009a. Dynamics in argumentation with single extensions: Abstraction principles and the grounded extension. In *Proc. of ECSQARU 2009*, 107–118.

Boella, G.; Kaci, S.; and van der Torre, L. 2009b. Dynamics in argumentation with single extensions: Attack refinement and the grounded extension. In *Proc. of AAMAS 2009*, 1213–1214.

Booth, R.; Kaci, S.; Rienstra, T.; and van der Torre, L. 2013. A logical theory about dynamics in abstract argumentation. In *Proc. of SUM 2013*. Springer. 148–161.

Cayrol, C.; de Saint-Cyr, F. D.; and Lagasquie-Schiex, M.-C. 2010. Change in abstract argumentation frameworks: Adding an argument. *Journal of Artificial Intelligence Research* 38:49–84.

Coste-Marquis, S.; Konieczny, S.; Mailly, J.-G.; and Marquis, P. 2014a. On the revision of argumentation systems: Minimal change of arguments statuses. In *Proc. of KR 2014*, 72–81.

Coste-Marquis, S.; Konieczny, S.; Mailly, J.-G.; and Marquis, P. 2014b. A translation-based approach for revision of argumentation frameworks. In *Proc. of JELIA 2014*, 77–85.

Coste-Marquis, S.; Konieczny, S.; Mailly, J.-G.; and Marquis, P. 2015. Extension enforcement in abstract argumentation as an optimization problem. In *Proc. of IJCAI 2015*, 2876–2882.

Dalal, M. 1988. Investigations into a theory of knowledge base revision: Preliminary report. In *Proc. of AAAI 1988*, 475–479.

Diller, M.; Haret, A.; Linsbichler, T.; Rümmele, S.; and Woltran, S. 2015. An extension-based approach to belief revision in abstract argumentation. In *Proc. of IJCAI 2015*, 2926–2932.

Doutre, S.; Herzig, A.; and Perrussel, L. 2014. A dynamic logic framework for abstract argumentation. In *Proc. of KR 2014*, 62–71.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence* 77(2):321–357.

Efstathiou, V., and Hunter, A. 2008. Algorithms for effective argumentation in classical propositional logic : A connection graph approach. In *Proc. of FoIKS 2008*, 272–290.

Efstathiou, V., and Hunter, A. 2011. Algorithms for generating arguments and counterarguments in propositional logic. *International Journal of Approximate Reasoning* 52(6):675–704.

Hamming, R. W. 1950. Error detecting and error correcting codes. *Bell System Technical Journal* 29(2):147–160.

Hunter, A. 2007. Real arguments are approximate arguments. In *Proc. of AAAI'07*, 66–71.

Katsuno, H., and Mendelzon, A. O. 1991. Propositional knowledge base revision and minimal change. *Artificial Intelligence* 52:263–294.

Katsuno, H., and Mendelzon, A. O. 1992. On the difference between updating a knowledge base and revising it. In Gärdenfors, P., ed., *Belief Revision*. 183–203.

Leite, J., and Martins, J. 2011. Social abstract argumentation. In *Proc. of IJCAI 2011*, 2287–2292.

Nouioua, F., and Würbel, E. 2014. Removed set-based revision of abstract argumentation frameworks. In *Proc. of ICTAI'14*, 784–791.

Thimm, M., and Villata, S. 2015. First International Competition on Computational Models of Argumentation (ICCMA'15). see `http://argumentationcompetition.org/2015/`.

Wallner, J. P.; Niskanen, A.; and Järvisalo, M. 2015. Complexity results and algorithms for extension enforcement in abstract argumentation. In *Proc. of AAAI'15*.

Iterated Ontology Revision by Reinterpretation

Özgür L. Özçep
Institute of Information Systems (IFIS)
University of Lübeck, Germany
oezcep@ifis.uni-luebeck.de

Abstract

Iterated applications of belief change operators are essential for different scenarios such as that of ontology evolution where new information is not presented at once but only in piecemeal fashion within a sequence. I discuss iterated applications of so called reinterpretation operators that trace conflicts between ontologies back to the ambiguous of symbols and that provide conflict resolution strategies with bridging axioms. The discussion centers on adaptations of the classical iteration postulates according to Darwiche and Pearl. The main result of the paper is that reinterpretation operators fulfill the postulates for sequences containing only atomic triggers. For complex triggers, a fulfillment is not guaranteed and indeed there are different reasons for the different postulates why they should not be fulfilled in the particular scenario of ontology revision with well developed ontologies.

1 Introduction

Iterated applications of belief change operators are essential for different scenarios such as that of ontology evolution where new information is not presented at once but only in piecemeal fashion within a sequence. Ontology evolution is a form of ontology change (Flouris et al. 2008) where an ontology modification is triggered by changes in the domain or in the conceptualization. The response to the change is the application of (a set of) predefined operators.

In this paper I consider the special scenario where an ontology (called the receiver's ontology) has to be changed along the arrival of a sequence of triggering bits of ontology fragments coming from another ontology (sender's ontology). In the terminology of Flouris et al. (Flouris et al. 2008) the one-step change would be termed ontology merge as the purpose is to get a better understanding of the domain from merging two ontologies over the same domain. As in our setting the merge is directed I call the kind of change operation *iterated ontology revision.*

One instance of iterated ontology revision is given by iterated reinterpretation operators (Eschenbach and Özçep 2010). In these operators, conflicts between the trigger and the receiver's ontology is explained by ambiguous use of terms. Consider an example of two online library systems with ontologies: the sender may use $Article$ to denote publications in journals whereas the receiver may use $Article$ to denote publications in journals or proceeding volumes. As the ontologies are assumed to be over the same domain, the receiver guesses relations on relations between her and the sender's uses and stipulates them as bridging axioms, e.g., stating that all articles in the sender's sense are articles in the receiver's sense.

Now, a challenging aspect is to define adequateness criteria that iterated ontology revision operators should fulfill. I consider the classical iteration postulates of Darwiche and Pearl (Darwiche and Pearl 1994) as possible candidates and state whether they are fulfilled by the reinterpretation operators. Moreover I discuss, for each of them, whether it should be fulfilled at all. The main result of the paper is that reinterpretation operators fulfill the postulates sequences with atomic triggers. For sequences of complex triggers, a fulfillment is not guaranteed and indeed there are different reasons—corresponding to different postulates—why they should not be fulfilled in the particular scenario of ontology revision with well developed ontologies.

The rest of the paper is structured as follows. After some logical preliminaries and general terminology (Sect. 2) the necessary definitions for reinterpretation operators are recapitulated (Sect. 3). Before the sections on related work and the conclusion, the adapted postulates for iterated revision, results on their fulfillment by reinterpretation operators, and a discussion of the results are given in Sect. 4.

2 Terminology and Logical Preliminaries

The reinterpretation framework described in the following works for any FOL theory but we consider here finite knowledge bases formulated in description logics.

A non-logical DL vocabulary consists of concept symbols (= atomic concepts) N_C, role symbols N_R, and individual constants N_i. Using these, more complex concept descriptions can be built up in a recursive fashion. The set of possible concept constructors depends on the specific DL. We consider in particular the basic constructors $\sqcap, \sqcup, \neg, \exists$. $\mathcal{C}(\mathcal{V})$ is the set of all possible concept descriptions that can be built from the symbols in $\mathcal{V}$ in the given description logic.

The semantics is the usual Tarskian semantics based on interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with a domain $\Delta^{\mathcal{I}}$ and denotation function $\cdot^{\mathcal{I}}$ which gives for every $c \in N_i$ an element $c^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, for every atomic concept $A \in N_C$ a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and for every role symbol $R \in N_R$ a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The denotation function is extended recursively to all concept descriptions in the usual manner. For the ones we use here, we have $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$; $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$; $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$; $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \text{There is } y \in C^{\mathcal{I}} \text{ s.t. } (x, y) \in R^{\mathcal{I}}\}$. Here $C, D \in \mathcal{C}(\mathcal{V})$ and $R \in N_R$. From concept descriptions one can built axioms which can be evaluated as true in (satisfied by) or false in (satisfied by) an interpretation. We consider TBox (terminological Box) axioms of the form

- $C \sqsubseteq D$ (concept subsumption) for $C, D \in \mathcal{C}(\mathcal{V})$ with semantics: $\mathcal{I} \models C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$;
- $R_1 \sqsubseteq R_2$ (role subsumption) for $R_1, R_2 \in N_R$ with semantics $\mathcal{I} \models R_1 \sqsubseteq R_2$ iff $(R_1)^{\mathcal{I}} \subseteq (R_2)^{\mathcal{I}}$

Moreover, we consider ABox axioms (assertional axioms) of the form $C(a)$ and $R(a, b)$ for $C \in \mathcal{C}(\mathcal{V})$, $a, b \in N_i$ and $R \in N_R$. The semantics is $\mathcal{I} \models C(a)$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $\mathcal{I} \models R(a, b)$ iff $(a, b) \in R^{\mathcal{I}}$. We call ABox axioms of the form $A(a)$ and $\neg A(a)$ with $A \in N_i$ concept assertions or concept-based literals. $\hat{A}$ stands for A or $\neg A$. Additionally, equalities $a \doteq b$, $a, b \in N_i$ may be allowed.

Consistency (= satisfiability) of a set of axioms X means that there is an interpretation $\mathcal{I}$ making all axioms in X true, for short $\mathcal{I} \models X$. Entailment is defined as usual by $O_1 \models O_2$ iff for all $\mathcal{I}$: If $\mathcal{I} \models O_1$, then $\mathcal{I} \models O_2$. A consequence operator Cn gives the set of all axioms following from a set: $\mathrm{Cn}(X) = \{ax \mid X \models ax\}$. If necessary, one can specify the vocabulary of the axioms: $\mathrm{Cn}^{\mathcal{V}}(X)$ is the set of axioms over $\mathcal{V}$ following from X. $X \equiv^{\mathcal{V}} Y$ is shorthand for $\mathrm{Cn}^{\mathcal{V}}(X) = \mathrm{Cn}^{\mathcal{V}}(Y)$. By $\mathcal{V}(O)$ we denote all non-logical symbols occurring in the set of axioms O. $\mathcal{C}(O) = \mathcal{C}(\mathcal{V}(O))$.

The ontology notion of this paper slightly extends the one known from the semantic web and DL community—the extension relying on the distinction between an internal vocabulary $\mathcal{V}'$ and a public vocabulary $\mathcal{V}$:

Definition 1 *An ontology $\mathcal{O}$ is a triple $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ consisting of a set of axioms O over a logic with non-logical symbols $\mathcal{V}$ (the public vocabulary) and $\mathcal{V}'$ (the internal vocabulary).*

In the following I will abuse terminology by calling also the set of axioms O ontology.

Let $O_1 \top O_2$ be the *dual remainder sets modulo O_2* (Delgrande 2008). This is the set of inclusion maximal subsets X of O_1 that are consistent with O_2, i.e., $X \in O_1 \top O_2$ iff $X \subseteq O_1$, $X \cup O_2$ is consistent and for all $Y \subseteq O_1$ with $X \subsetneq Y$ the set $Y \cup O_2$ is not consistent.

We are going to deal with substitutions as means to realize name space dissociations. The set of *ambiguity compliant resolution substitutions*, denoted $\mathrm{AR}(\mathcal{V}, \mathcal{V}')$, consists of substitutions of symbols in $\mathcal{V}$ by symbols in $\mathcal{V} \cup \mathcal{V}'$. Here, we assume $\mathcal{V} \cap \mathcal{V}' = \emptyset$ where $\mathcal{V}'$ is the set of symbols used for internalization. The substitutions in $\mathrm{AR}(\mathcal{V}, \mathcal{V}')$ get as input a non-logical symbol in $\mathcal{V}$ (a constant, an atomic concept or role in DL speak) and map it either to itself or to a new non-logical symbol (of the same type) in $\mathcal{V}'$. The set of symbols $s \in \mathcal{V}$ for which $\sigma(s) \neq s$ is called the *support* of σ and is denoted $\mathrm{sp}(\sigma)$. In the following I use postfix notation for substitutions, i.e., $X\sigma = \sigma(X)$. Moreover, I use the following shorthands $\mathrm{sp}_i(\sigma) = \mathrm{sp}(\sigma) \cap N_i$ and $\mathrm{sp}_{CR}(\sigma) = \mathrm{sp}(\sigma) \cap (N_C \cup N_R)$. A substitution with support S is also denoted by σ_S. For substitutions $\sigma_1, \sigma_2 \in \mathrm{AR}(\mathcal{V}, \mathcal{V}')$ we define an ordering by: $\sigma_1 \leq \sigma_2$ iff $\mathrm{sp}(\sigma_1) \subseteq \mathrm{sp}(\sigma_2)$. $\mathrm{AR}(\mathcal{V}, \mathcal{V}')$ can be partitioned into equivalence classes of substitutions that have the same support. We assume that for every equivalence class a representative substitution $\Phi(S) \in \mathrm{AR}(\mathcal{V}, \mathcal{V}')$ with support S is fixed. Φ is called a *disambiguation schema*.

3 Reinterpretation Operators

This section recapitulates the definitions of ontology revision operators called reinterpretation operators (Eschenbach and Özçep 2010; Özçep 2008). The envisioned scenario is that of two agents holding well-developed ontologies, one called receiver's ontology, the other called sender's ontology. The ontologies are over the same domain and the receiver gets bits of information from the sender's ontology that she wants to integrate into her ontology in order to get a better, more fine-grained model of the domain. A challenging aspect is to preserve the consistency of the ontology. The kind of inconsistency that is considered here is that of inter-ontological ambiguity: the sender and the receiver may use the same symbol with different meanings (compare for example the different uses of $Article$ in the example below).

So, the conflict resolution strategy that is exploited by the reinterpretation operators is based on disambiguating symbols. The sender or the receiver has to reinterpret an ambiguous symbol. In the more interesting non-monotonic setting, that I consider in this paper, it is always the receiver who reinterprets the ambiguous symbol—by storing the old symbol in a new name space and relating her use of the symbols to the sender's use by bridging axioms. This is in line with classical (prioritized) belief revision where one has full trust in the trigger information. In (Eschenbach and Özçep 2010) these reinterpretation operators are called type-2 operators, contrasting them with type-1 operators in which it is the sender's terminology that is reinterpreted.

The weak reinterpretation operators $\otimes$ and strong reinterpretation operators $\odot$ are binary operators with a finite set of ontology axioms ontology as left and right argument. The following example (Özçep 2012) demonstrates the main ideas for the weak reinterpretation operators.

Example 1 *Consider the sets of ontology axioms O_1, O_2 of the receiver and sender, resp.:*

$$\begin{aligned} O_1 &= \{Article(pr_1), Article(pr_2), \neg Article(bo_1)\} \\ O_2 &= \{\neg Article(pr_1)\} \end{aligned}$$

Applying the weak reinterpretation operator $\otimes$ gives the following set of axioms:

$$\begin{aligned} O_1 \otimes O_2 \quad = \quad & \{Article'(pr_1), Article'(pr_2), \\ & \neg Article'(bo_1), \neg Article(pr_1), \\ & Article \sqsubseteq Article'\} \end{aligned}$$

For the purpose of the example I assume that only concept and role symbols but not constant symbols may be used ambiguously. So, the above conflict between the sender's and *receiver's ontology can only be caused by different uses of the atomic concept $Article$. The receiver (holder of O_1) gives priority to the sender's use of $Article$ over her use of $Article$, and hence she adds $\neg Article(pr_1)$ into the result $O_1 \otimes O_2$. The receiver's use of $Article$ is internalized, i.e., all occurrences of $Article$ in O_1 are substituted by a new symbol $Article'$. This step of internalization will also be called the step of dissociation or disambiguation. Additionally, the receiver adds hypotheses on the semantical relatedness (bridging axioms) of her and the sender's use of $Article$, here $Article \sqsubseteq Article'$ which states that $Article$ is a subconcept of $Article'$.*

Technically, the disambiguation is realized by uniform substitutions from $\mathrm{AR}(\mathcal{V}, \mathcal{V}')$ (see section on logical preliminaries). For the disambiguation, one has to deal with the a potential multiplicity of conflicts. The minimal conflict symbol sets defined below describe the smallest sets of symbols which have to be disambiguated in order to resolve conflicts.

Definition 2 *For ontologies O_1, O_2 over $\mathcal{V}$ the set of* minimal conflicting symbols sets, $\mathrm{MCS}(O_1, O_2)$, *is defined as follows:*

$$\begin{aligned}&\mathrm{MCS}(O_1, O_2) = \\ &\quad \{\, S \subseteq \mathcal{V} \mid \textit{There is a } \sigma_S \in \mathrm{AR}(\mathcal{V}, \mathcal{V}'),\ \textit{s.t.} \\ &\qquad\qquad O_1\sigma_S \cup O_2 \textit{ is consistent, and for} \\ &\qquad\qquad \textit{all } \sigma_{S_1} \in \mathrm{AR}(\mathcal{V}, \mathcal{V}') \textit{ with } \sigma_{S_1} < \sigma_S \\ &\qquad\qquad O_1\sigma_{S_1} \cup O_2 \textit{ is not consistent. } \}\end{aligned}$$

Following the strategy of AGM partial meet revision (Alchourrón, Gärdenfors, and Makinson 1985), we assume that a selection function γ_1 selects candidates from $\mathrm{MCS}(O_1, O_2)$ to be used for the resolution: $\gamma_1(\mathrm{MCS}(O_1, O_2)) \subseteq \mathrm{MCS}(O_1, O_2)$. So the symbol set defined by $S^{\#} = \bigcup \gamma_1(\mathrm{MCS}(O_1, O_2))$ is the set of symbols which will be internalized.

To regain as much as possible from the receiver's ontology in the ontology revision result, the disambiguated symbols of $S^{\#}$ are related by bridging axioms. Depending on what kind of bridging axioms are chosen, different revision operators result. In this paper we consider two classes of bridging axioms, the *simple bridging axioms* and the *strong bridging axioms* (Özçep 2008). Let $\sigma = \sigma_S \in \mathrm{AR}(\mathcal{V}, \mathcal{V}')$ be a substitution with support $S \subseteq \mathcal{V}$. Let P be a concept or role symbol in S, $\sigma(P) = P'$.

Definition 3 *Let $\sigma = \sigma_S \in \mathrm{AR}(\mathcal{V}, \mathcal{V}')$ for $S \subseteq \mathcal{V} \cap (N_C \cup N_R)$. The set of* simple bridging axioms *w.r.t. σ is*

$$\mathcal{B}(\sigma) = \{P \sqsubseteq P', P' \sqsubseteq P \mid P \in S\}$$

The set of strong bridging axioms *w.r.t. σ is defined as:*

$$\begin{aligned}&\dot{\mathcal{B}}(\sigma, O) = \\ &\quad \{C\sigma \sqsubseteq s \mid C \in \mathcal{C}(O), s \in \mathrm{sp}_{CR}(\sigma), O \models C \sqsubseteq s\} \cup \\ &\quad \{s \sqsubseteq C\sigma \mid C \in \mathcal{C}(O), s \in \mathrm{sp}_{CR}(\sigma), O \models s \sqsubseteq C\} \cup \\ &\quad \{s \doteq s\sigma \mid s \in \mathrm{sp}_i(\sigma)\}\end{aligned}$$

In case of conflict, not all bridging axioms of $\mathcal{B}(S^{\#})$ (resp. $\dot{\mathcal{B}}(S^{\#})$) can be added to the integration result (compare Ex. 1). Hence, one searches for subsets that are compatible with the union of the internalized ontology and sender ontology, $O_1\sigma \cup O_2$. That means, possible candidate sets of bridging axioms can be described by dual remainder sets (see section on logical preliminaries) as $\mathcal{B}(\sigma)\top(O_1\sigma \cup O_2)$. Again, as there is no preference for one candidate over the other we assume that a second selection function γ_2 is given with $\gamma_2(\mathcal{B}(\sigma)\top(O_1\sigma \cup O_2)) \subseteq (\mathcal{B}(\sigma)\top(O_1\sigma \cup O_2)$. The intersections of the selected bridging axioms is the set of bridging axioms added to the integration result. (Compare this with the partial meet revision functions of AGM (Alchourrón, Gärdenfors, and Makinson 1985)).

Definition 4 *Let $\mathcal{V}, \mathcal{V}'$ be disjoint vocabularies and Φ a disambiguation scheme. Moreover let γ_1, γ_2 be selection functions and for short let $\overline{\gamma} = (\gamma_1, \gamma_2)$. For any ontology O_1 and O_2 over $\mathcal{V}$ let $S^{\#} = \bigcup \gamma_1(\mathrm{MCS}(O_1, O_2))$ and $\sigma = \Phi(S^{\#})$. Then the* weak reinterpretation operator $\otimes^{\overline{\gamma}}$ *and the* strong reinterpretation operator $\odot^{\overline{\gamma}}$ *are defined as follows:*

$$\begin{aligned}O_1 \otimes^{\overline{\gamma}} O_2 &= O_1\sigma \cup O_2 \cup \bigcap \gamma_2\big(\mathcal{B}(\sigma)\top(O_1\sigma \cup O_2)\big) \\ O_1 \odot^{\overline{\gamma}} O_2 &= O_1\sigma \cup O_2 \cup \\ &\qquad \bigcap \gamma_2\big(\dot{\mathcal{B}}(\sigma, O_1)\top(O_1\sigma \cup O_2)\big)\end{aligned}$$

The definition for weak reinterpretation operators is the same as in (Özçep 2008), the definition of the strong operators is an extension.

In the following I will simplify the discussion by simplifying the first step of internalization: In the internalization step now all symbols of the receiver are internalized. There is only a selection function for bridging axioms. Due to the fact that the maximal candidates of bridging axioms are used in the reinterpretation operators unnecessary internalizations do not occur. I therefore can write in the following, e.g., $\otimes^{\gamma}$ instead of $\otimes^{\overline{\gamma}}$.

A particularly interesting case of ontology change appears in the context of ABox update (Ahmeti, Calvanese, and Polleres 2014; Gutierrez, Hurtado, and Vaisman 2011), where the trigger informations are assertional axioms (ABox) axioms. I consider the special case that only atomic bits from the sender ontology occur as trigger, namely, the trigger O_2 is of the form $O_2 = \{A(a)\}$ or of the form $O_2 = \{\neg A(a)\}$. That is, the trigger is a concept assertion with an atomic symbol ($A \in N_C$) or the negated atomic symbols $\neg A$. For this special case particular strong reinterpretation operators can be defined (Eschenbach and Özçep 2010). The first class assumes that within the underlying DL a most specific concept w.r.t. an ontology exists. C is a most specific concept for b in the ontology O iff $O \models C(b)$ and for all C' s.t. $O \models C'(b)$ also $O \models C \sqsubseteq C'$. The most specific concept is unique modulo concept equivalence (w.r.t. O), hence it is denoted by $\mathrm{msc}_O(b)$.

Definition 5 *Let $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ be an ontology, Φ a disambiguation scheme, $A \in \mathcal{V} \cap N_C$ and $b \in \mathcal{V} \cap N_i$. Assume $\sigma = [A/A']$ is the substitution fixed by Φ and assume that* $\mathrm{msc}_O(b)$ *exists. The* msc-based strong reinterpretation operators for concept-based literals $\boxdot$*) are defined as follows: If $O \cup \{A(b)\} \not\models \bot$ let $O \boxdot A(b) = O \cup \{A(b)\}$. Else:*

$$\begin{aligned}&O \boxdot A(b) = \\ &\quad \sigma(O) \cup \{A(b), A' \sqsubseteq A, A \sqsubseteq A' \sqcup \mathrm{msc}_{O\sigma}(b)\}\end{aligned}$$

If $O \cup \{\neg A(b)\} \not\models \bot$, *let* $O \boxdot \neg A(b) = O \cup \{\neg A(b)\}$. *Else:*

$$\begin{aligned} &O \boxdot \neg A(b) = \\ &\quad \sigma(O) \cup \{\neg A(b), A \sqsubseteq A', A' \sqsubseteq A \sqcup \mathrm{msc}_{O\sigma}(b)\} \end{aligned}$$

The following examples illustrates the second case.

Example 2 *Assume that the receiver's ontology from the beginning is extended with two additional facts on the "problematic" entity* pr_1*:*

$$O_1^+ = O_1 \cup \{\mathit{publishedIn}(pr_1, proc1), \mathit{Proceed}(proc1)\}$$

The most specific concept of pr_1 *w.r.t.* O_1^+ *is*

$$\mathrm{msc}_{O_1^+}(pr_1) = \mathit{Article} \sqcap \exists \mathit{publishedIn}.\mathit{Proceed}$$

Hence the result of strong reinterpretation w.r.t. triggering concept assertions $O_1^+ \boxdot \neg \mathit{Article}(pr_1)$ *adds the following additional bridging axiom*

$$\mathit{Article}' \sqsubseteq \mathit{Article} \sqcup (\mathit{Article}' \sqcap \exists \mathit{publishedIn}.\mathit{Proceed})$$

This says that the wider use of Article by the receiver adds (only) those publications in proceedings into the extension.

The selection-based strong operators for triggering literals provide more bridging axioms between the internalized and non-internalized symbols.

Definition 6 *Let* $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ *be an ontology,* Φ *a disambiguation scheme,* $A \in \mathcal{V} \cap N_C$ *and* $b \in \mathcal{V} \cap N_i$. *Assume* $\sigma = [A/A']$ *is the substitution fixed by* Φ *and that* $\mathrm{msc}_O(b)$ *exists. Moreover, let* sel *be an arbitrary selection function, defined as* $\mathrm{sel}(X) \subseteq X$. *The* selection-based strong reinterpretation operators for concept-based literals $\oplus^{\mathrm{sel}}$*) are defined as follows (using auxiliary definitions for the specific bridging axioms):*

$$\begin{array}{rcl} \mathrm{oa}(O, A(b), K') & = & \{A \sqsubseteq A' \sqcup C \mid C \in \mathcal{C}(\mathcal{V} \cup \mathcal{V}'), \\ & & \quad O\sigma \models C(b) \text{ and } A \notin \mathcal{V}(C)\} \\ \mathrm{oa}(O, \neg A(b), A') & = & \{A' \sqsubseteq A \sqcup C \mid C \in \mathcal{C}(\mathcal{V} \cup \mathcal{V}'), \\ & & \quad O\sigma \models C(b) \text{ and } A \notin \mathcal{V}(C)\} \\ O \oplus^{\mathrm{sel}} \alpha & = & \begin{cases} O \cup \{\alpha\} & \text{if } O \cup \{\alpha\} \not\models \bot \\ O \otimes \{\alpha\} \cup \mathrm{sel}(\mathrm{oa}(O, \alpha, A')) & \\ \quad \text{else} & \end{cases} \end{array}$$

Though the complexity of the trigger is low the induced concept lattice for the reinterpretation with $\oplus^{\mathrm{sel}}$ is not trivial as illustrated by Fig. 1. Nonetheless, the figure does not suggest that the computation of the revision outcome is more complex than for other revision operators for DL ontologies: It just illustrates the subsumption connections of the concepts within the resulting ontology; the calculation of the lattice is not part of constructing the revision result.

Without proof I state here some observations on the conservativity of reinterpretation operators. Proofs can be found in (Eschenbach and Özçep 2010).

Proposition 1 *Let* $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ *be an ontology,* Φ *a disambiguation scheme,* $A \in \mathcal{V} \cap N_C$ *and* $a, c \in \mathcal{V} \cap N_i$. *Assume* $\sigma = [A/A']$ *is the substitution fixed by* Φ *and that* $\mathrm{msc}_O(a)$ *exists. Let* $\alpha = A(a)$ $\epsilon = A(c)$ *or* $\alpha = \neg A(a)$ *and* $\epsilon = \neg A(c)$. *Let* β *be an assertion with* $\mathcal{V}(\beta) \subseteq (\mathcal{V} \cup \mathcal{V}(O)) \setminus \{A\}$. *Let* sel *be a selection function for bridging axioms and* $\widetilde{\mathrm{sel}}$ *a corresponding function selecting corresponding concepts:* $\widetilde{\mathrm{sel}}(\mathrm{oa}(O, \hat{A}(a), A')) = \{C \mid \hat{A} \sqsubseteq \hat{A}' \sqcup C \in \mathrm{sel}(\mathrm{oa}(O, \hat{A}(a), A'))\}$.

If $O \models \neg\alpha$, *then*

$$\begin{array}{rrlll} 1. & O \circ \alpha & \models \beta & \textit{iff} & O \models \beta \\ 2. & O \circ \alpha & \models \epsilon & \textit{iff} & O \cup \{a \not\doteq c\} \models \epsilon \\ 3. & O \otimes \alpha & \not\models \neg\epsilon & & \\ 4. & O \boxdot \alpha & \models \neg\epsilon & \textit{iff} & O \models \neg\epsilon \textit{ and} \\ & & & & O \models \neg\, \mathrm{msc}_O(a)(c) \\ 5. & O \oplus^{\mathrm{sel}} \alpha & \models \neg\epsilon & \textit{iff} & O \models \neg\epsilon \textit{ and} \\ & & \multicolumn{3}{l}{O \models \neg \sqcap \widetilde{\mathrm{sel}}(\mathrm{oa}_2(O, \alpha, A'))_{[A'/A]}(c)} \end{array}$$

4 Postulates for Iterated Reinterpretation

Many forms of ontology change (Flouris et al. 2008), in particular ontology evolution (Kharlamov, Zheleznyakov, and Calvanese 2013), require the iterated application of a change operator under new bits of informations. In iterated belief revision, this problem is approached systematically by defining, both, postulates and operators for iterated applications of revision operators. The first systematic study of iterated belief revision goes back to the work of Darwiche and Pearl (Darwiche and Pearl 1994) who stressed the fact that the AGM postulates (Alchourrón, Gärdenfors, and Makinson 1985) are silent w.r.t. the iterated application of operators. Indeed, the only postulates that can be said to touch some form of iteration are those dealing with the revision of conjunctions of triggers (supplementary postulates 7 and 8).

I state those postulates in a form adapted to ontologies.

(RAGM 7) $\mathrm{Cn}^{\mathcal{V}}(O \circ (O_1 \cup O_2)) \subseteq \mathrm{Cn}^{\mathcal{V}}((O \circ O_1) \cup O_2)$

Postulate (RAGM 7) says that all sentences over $\mathcal{V}$ following from $O \circ (O_1 \cup O_2)$ are contained in the revision by O_1 followed by an expansion with O_2.

(RAGM 8) If $(O \circ O_1) \cup O_2 \not\models \bot$, then :

$$\mathrm{Cn}^{\mathcal{V}}((O \circ O_1) \cup O_2) \subseteq \mathrm{Cn}^{\mathcal{V}}(O \circ (O_1 \cup O_2))$$

Postulate (RAGM 8) says that all sentences over $\mathcal{V}$ following from the result of revising with O_1 and expanding with O_2 also follow from revising O with the union of O_1 and O_2. A precondition is that the revision result by O_1 is compatible with O_2.

In general, the reinterpretation operators do not fulfill these postulates. This can be shown with examples similar those provided by Delgrande and Schaub (Delgrande and Schaub 2003, p. 13).

But if one chooses particular selection functions for the reinterpretation operators on triggering ontologies, then one can show that the postulates (RAGM 7) and (RAGM 8) are fulfilled. This result is similar to an AGM theorem (Alchourrón, Gärdenfors, and Makinson 1985) which says that a partial meet revision operator on belief sets fulfills all AGM postulates (in particular the supplementary ones) iff it can be defined as a transitive relational partial meet revision operator.

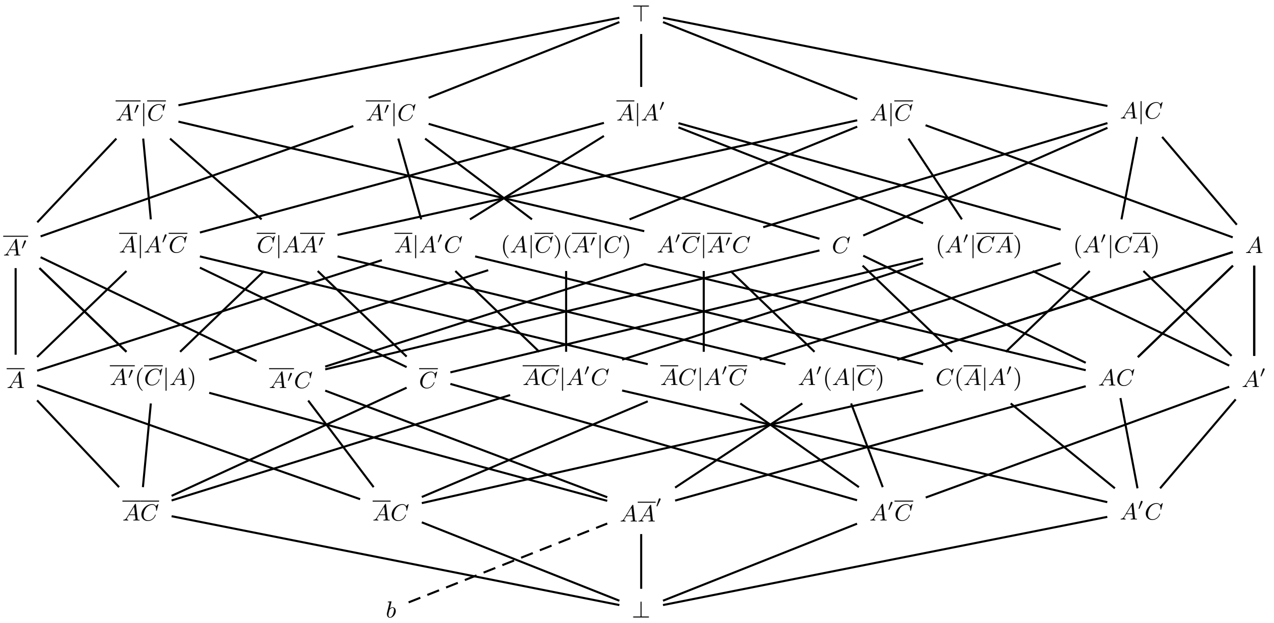


Figure 1: Concept lattice for $O \oplus^{\text{sel}} A(b)$ for the case $O \models \neg A(b)$. We assume that the set of concepts chosen by sel is representable as a concept description C. Here, $A_1 \sqcap A_2$ is abbreviated as A_1A_2, $\neg A$ by $\overline{A}$ and $A_1 \sqcup A_2$ by $A_1|A_2$

Definition 7 *A selection function γ for bridging axioms is called* a maximum based selection function for bridging axioms *iff the following holds:*

1. $|\gamma(X)| = 1$ *for all* $\emptyset \neq X \subseteq \mathcal{B}(\sigma_{\mathcal{V}})$.
2. *If BA_1 and BA_2 are non-empty sets of bridging axioms from $\mathcal{B}(\sigma_{\mathcal{V}})$, i.e., $BA_1, BA_2 \in \text{Pow}(\mathcal{B}(\sigma_{\mathcal{V}})) \setminus \{\emptyset\}$ s.t. for all $X_2 \in BA_2$ there is a $X_1 \in BA_1$ with $X_2 \subseteq X_1$, and if additionally also $\gamma(BA_1) \subseteq BA_2$ holds, then $\gamma(BA_2) = \gamma(BA_1)$.*

Now one can show

Proposition 2 *Let $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$, $\mathcal{O}_1 = (O_1, \mathcal{V}, \emptyset)$ and $\mathcal{O}_2 = (O_2, \mathcal{V}, \emptyset)$ be ontologies and γ be a maximum based selection function for bridging axioms. Then:*
If $(O \otimes^{\gamma} O_1) \cup O_2$ is consistent, then : $O \otimes^{\gamma} (O_1 \cup O_2) = (O \otimes^{\gamma} O_1) \cup O_2$.

This proposition shows that weak reinterpretation operators with maximum based selection function fulfill (RAGM 8). Moreover, one sees immediately that they fulfill the postulate (RAGM 7) because if $(O \otimes^{\gamma} O_1) \cup O_2$ is inconsistent, then trivially:

$$\text{Cn}^{\mathcal{V}}(O \otimes^{\gamma} (O_1 \cup O_2)) \subseteq \text{Cn}^{\mathcal{V}}((O \otimes^{\gamma} O_1) \cup O_2)$$

The supplementary postulates do not give constraints for the interesting case where for both triggers genuine revisions have to be applied. This motivated Darwiche and Pearl (Darwiche and Pearl 1994) to define four iteration postulates which, in an adaptation for the ontology revision scenario, are investigated in the following. The results below show that the reinterpretation operators in general do not fulfill the postulates.

The postulates are, along the original ideas of Darwiche and Pearl (Darwiche and Pearl 1994), described for finite sets of sentences (here: ontology axioms), and not for epistemic states as in their follow-up paper (Darwiche and Pearl 1997). Using the terminology of Freund and Lehmann (Freund and Lehmann 2002), the type of iterated revision I consider in this paper is *static*: There is no (used) encoding of the revision history in an epistemic state nor do I consider the dynamic change of revision operators from step to step. As we consider the justification of the iteration postulates not as a whole but one by one this approach does not stand in contradiction to the insights made in the follow-up paper by Darwiche and Pearl (Darwiche and Pearl 1997).

In the following postulates, $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ is the initial ontology, $\mathcal{O}_1 = (O_1, \mathcal{V}, \emptyset)$ the first triggering ontology and $\mathcal{O}_2 = (O_2, \mathcal{V}, \emptyset)$ the second triggering ontology. Note that the trigger ontologies do not contain internal symbols—which fits the idea that only the public parts of the sender ontologies are communicated.

(RDP 1) If $O_2 \models O_1$, then $(O \circ O_1) \circ O_2 \equiv_{\mathcal{V}} O \circ O_2$.

In natural language: If the axioms of the second trigger ontology are stronger than the ones of the first trigger ontology, then the two-step outcome (relativized to the public vocabulary) is already covered by the revision with the second trigger ontology.

(RDP 2) If $O_1 \cup O_2$ is not consistent,
then $(O \circ O_1) \circ O_2 \equiv_{\mathcal{V}} O \circ O_2$.

In natural language: If the axioms of the first and second trigger ontology are incompatible, then the two-step outcome (relativized to the public vocabulary) is already covered by the revision with the second trigger ontology.

(RDP 3) If $O \circ O_2 \models O_1$, then $(O \circ O_1) \circ O_2 \models O_1$.

In natural language: If the revision by the second trigger ontology entails the first trigger ontology, then the entailment still holds for the revision with the first ontology followed by the second trigger ontology.

(RDP 4) If $O_1 \cup (O \circ O_2)$ is consistent
then so is $O_1 \cup (O \circ O_1) \circ O_2$.
In natural language: If the revision by the second trigger ontology is compatible with the first trigger ontology, then the compatibility still holds for the revision with the first followed by the second trigger ontology.

As the following Proposition 3 shows, the fulfillment of all adapted iteration postulates cannot be guaranteed if the trigger is an ontology. (This is the same as for the operators of Delgrande and Schaub (Delgrande and Schaub 2003).) If the trigger is of atomic nature the situation is different due to the fact that there is only one symbol to be reinterpreted. For triggering literals only (RDP 2) is not fulfilled.

Proposition 3 states results for all reinterpretation operators mentioned in this paper: Regarding the weak operators a distinction is made between triggering literals and ontologies. Table 1 summarizes the results. The rows contain the operators: The first three having concept-based literals as triggers, the last two ontologies. The columns except for the last one refer to the iteration postulates. The last column gives a reference to the corresponding result in Proposition 3. Regarding the counterexamples I draw the following distinction—also reflected in the table: The weak counterexamples are those that construct ontologies for a specific selection function. The strong counterexamples are those that construct ontologies for any selection function.

In the counter examples that were used to prove the negative results all reinterpretation operators reinterpret only atomic concepts and roles but not constants. As long as a conflict resolution by reinterpreting only concepts and roles is possible, the reinterpretation operators can be modeled by a suitable definition of a selection function γ^{CR}: γ^{CR} selects only sets of bridge axioms that contain all identities for all constants which, in the end, means that the constants are not reinterpreted. In this case I call γ^{CR} a *selection function that prefers the reinterpretation of role and concept symbols.*

Proposition 3 *Regarding the fulfillment of the adapted iteration postulates of Darwiche and Pearl (Darwiche and Pearl 1997) the following results hold.*

1. *Reinterpretation operators for concept-based triggers* $(\otimes,\boxdot,\oplus^{\mathrm{sel}})$ *fulfill (RDP 1), (RDP 3) and (RDP 4).*
   There are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ such that $\otimes, \boxdot$ and $\oplus^{\mathrm{sel}}$ (for all selection functions sel*) do not fulfill (RDP 2).*
   There are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ and a selection function sel *such that $\oplus^{\mathrm{sel}}$ does not fulfill (RDP 3) and does not fulfill (RDP 4).*
2. *For weak reinterpretation operators over triggering ontologies $\otimes^\gamma$ the following holds:*
   For all postulates (RDP x), $1 \leq x \leq 3$, there are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ such that for all selection functions γ^{CR} that prefer the reinterpretation of role and concept symbols $\otimes^\gamma$ does not fulfill (RDP x).
   There are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ and a selection function γ^{CR} that prefer the reinterpretation of role and concept symbols such that $\otimes^\gamma$ does not fulfill (RDP 4).
3. *For strong reinterpretation operators over triggering ontologies $\odot^\gamma$ the following holds:*
   For all postulates (RDP 1), (RDP 3), (RDP 4) there are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ and selection functions γ such that $\odot^\gamma$ does not fulfill any of them.
   There are ontologies $\mathcal{O}, \mathcal{O}_1, \mathcal{O}_2$ such that for all selection functions γ^{CR} that prefer the reinterpretation of concepts and role symbols $\odot^\gamma$ does not fulfill (RDP 2).

I discuss the outcomes of the proposition for the four postulates one by one starting with (RDP 2) which (in its original form given by Darwiche and Pearl) evoked most of the criticism. Regarding this postulate I follow the argument of Delgrande and Schaub (Delgrande and Schaub 2003) according to which (RDP 2) may make sense only for non-complex triggers. For complex triggers, say in our case: complex ontologies O_1 (and O_2), does not work. Assume O_1 is made out of two sub-ontologies O_{11} und O_{12} s.t. only O_{12} is not compatible with O_2. All those assertions that follow from $O \circ O_1$ on the basis of O_{11} should be conserved after the revision with O_2. But according to (RDP 2) amnesic revision would be allowed if O_2 would not entail O_{11}: All sentences inferred with O_{11} would be eliminated in favor of the new ontology O_2.

Regarding the first iteration postulate, the following simple example by Delgrande and Schaub (Delgrande and Schaub 2003) demonstrates its questionable status. Actually, for the proof of Proposition 3.2 I use adapted variants of this example. Consider the ontologies $O = \{\neg A(a)\}$, $O_1 = \{(A \sqcup B)(a)\}$ und $O_2 = \{A(a)\} \models O_1$. Let γ be a selection function that prefers the reinterpretation of concept and role symbols. The second ontology O_2 ist stronger than the first ontology O_1. Revision with O_2 leads to an ontology in which $B(a)$ does not hold: $O \otimes^\gamma O_2 \not\models B(a)$. The revision with the first ontology leads to an ontology in which $B(a)$ holds: $O \otimes^\gamma O_1 = \{\neg A(a), (A \sqcup B)(a)\} \models B(a)$. The revision by the first and then by the second ontology gives an ontology that still entails $B(a)$:

$$\begin{aligned}(O \otimes^\gamma O_1) \otimes^\gamma O_2 \quad = \quad & \{\neg A(a), (A \sqcup B)(a), A'(a'), \\ & A \sqsubseteq A', a \doteq a', B \sqsubseteq B', \\ & B' \sqsubseteq B\} \quad \models \quad B(a)\end{aligned}$$

All preconditions in the antecedent of (RDP 1) are fulfilled but not the succedens: $(O \otimes^\gamma O_1) \otimes^\gamma O_2 \not\equiv^{\mathcal{V}} O \otimes^\gamma O_2$.

There is no plausible revision operator for this particular ontology setting that would fulfill (RDP 1). Such an operator would have to fulfill $O \circ O_1 \models B(a)$ as O and O_1 are compatible. The revision with O_2 should not eliminate $B(a)$ as $B(a)$ is not relevant for the conflict: $(O \circ O_1) \circ O_2 \models B(a)$. Clearly, one could define syntax-sensitive revision operators on belief bases s.t. $(O \circ O_1) \circ O_2 \not\models B(a)$ so that the fulfillment of (RDP 1) could be achieved also for this ontol-

Operator	(RDP 1)	(RDP 2)	(RDP 3)	(RDP 4)	Proposition 3.x
$\otimes$	+	–	+	+	1
$\oplus^{\mathrm{sel}}$	+	– $(\forall\,\mathrm{sel})$	+	+	
$\boxdot$	+	–	+	+	
$\otimes^{\gamma}$	– $(\forall\gamma^{\mathrm{CR}})$	– $(\forall\gamma^{\mathrm{CR}})$	– $(\forall\gamma^{\mathrm{CR}})$	– $(\exists\gamma^{\mathrm{CR}})$	2
$\odot^{\gamma}$	– $(\exists\gamma^{\mathrm{CR}})$	– $(\forall\gamma^{\mathrm{CR}})$	– $(\exists\gamma^{\mathrm{CR}})$	– $(\exists\gamma^{\mathrm{CR}})$	3

Table 1: Results of Proposition 3
A + entry means that the postulate is fulfilled for all ontologies. A – entry means that there is an ontology such that the postulate is not fulfilled (only used for triggering literals). An entry of type – $(\forall\,\mathrm{sel})$ resp. – $(\forall\gamma)$ resp. – $(\forall\gamma^{\mathrm{CR}})$ means that there are ontologies s.t. for all selection functions sel resp. γ resp. γ^{CR} the postulate is not fulfilled. An entry of type – $(\exists\gamma^{\mathrm{CR}})$ means, that there is a selection function γ^{CR} such that the postulate is not fulfilled.

ogy configuration. But this does not change the situation that also $(O \circ O_1) \circ O_2 \models B(a)$ should be fulfilled. Moreover, syntax-sensitive belief-base operators are not appropriate for the revision of ontologies for which we would like to ensure (unique) syntax insensitive representations. So the only possibility for $\circ$ to fulfill (RDP 1) is that $O \circ O_2 \models B(a)$ holds. Such an operator $\circ$ that fulfills these conditions can be defined : O entails $(\neg A \sqcup B)(a)$ and $(\neg A \sqcup \neg B)(a)$. If $\circ$ has a selection function γ that chooses $(\neg A \sqcup B)(a)$, then $O \circ O_2$ would entail $B(a)$. But one could equally have a selection function γ' such that $O \circ O_2 \not\models B(a)$ or even $O \circ O_2 \models \neg B(a)$. There is no adequate reason for assuming that one has to prefer γ over γ'.

The counter example against (RDP 1) refers to triggering ontologies. For non-complex triggers such as concept-based literals a counter example cannot be constructed. Indeed: In this case all reinterpretation operators fulfill (RDP 1) (besides (RDP 3) and (RDP 4) (see Proposition 3.1).

Regarding the weak reinterpretation operator for triggering ontologies $\otimes^{\gamma}$ one can construct examples such that for all selection functions γ that prefer the reinterpretation of concept and role symbols $\otimes^{\gamma}$ does not fulfill the postulate (RDP 3) (Proposition 3.2). For strong reinterpretation operators for triggering ontologies one can at least construct ontologies and at least one selection function showing the non-fulfillment (RDP 3). The counter example for the weak variants is based on the interplay of trivial revision (consistency case) and non-trivial revision (inconsistency case):

$$\begin{aligned} O &= \{A(a), \neg B(a) \vee \neg A(c), A(b) \vee \neg A(e)\} \\ O_1 &= \{\neg A(b)\} \\ O_2 &= \{\neg A(a), B(a), A(c) \vee \neg A(b), A(e)\} \end{aligned}$$

O and O_1 are chosen such that they are compatible and so $O \otimes^{\gamma} O_1 = O \cup O_1 \models \neg A(e)$. Due to the antecedens in postulate (RDP 3) the revision by the second triggering ontology O_2 gives an ontology $O \otimes^{\gamma} O_2$ that entails the first trigger O_1. The conflict resolution for $O_2 \cup O$ is such that O_1 is not effected by it. But a previous revision with O_1 requires a different (additional) conflict resolution with O_2 such that O_1 is not entailed anymore: $(O \otimes^{\gamma} O_1) \otimes^{\gamma} O_2 \not\models \{\neg A(b)\} (= O_1)$. The reason that $\neg A(b)$ cannot be inferred anymore is due to the fact that the conflict resolution for $O \otimes^{\gamma} O_1$ and O_2 leads to a reinterpretation of A, and due to the fact that the assertion $\neg A(e)$, which follows from $O \otimes^{\gamma} O_1$, has the same polarity as $\neg A(b)$: namely, it is also negated.

This lost of $\neg A(b)$ is due to the construction of the reinterpretation operators which implement a uniform reinterpretation: In case of conflicts all occurrences of symbols involved in the conflict are internalized. Only by introducing bridging axioms is it possible to regain assertions in the public vocabulary. But when the bridging axioms are not expressive enough, then old sentences of the receiver (such as $\neg A(b)$ in this example) may not be entailed anymore. This last discussion regarding (RDP 3) (and similarly for (RDP 4)) cannot be used as general arguments against (RDP 3) and (RDP 4) as adequate reinterpretation postulates. One may construct plausible ontology revision operators fulfilling (RDP 3) and (RDP 4), but these cannot implement uniform reinterpretation: They would have to do partial reinterpretation (as, e.g., done by Goeb and colleagues (Goeb et al. 2007)). So, acceptable arguments against (RDP 3) and (RDP 4) would have to support the requirement of uniformity within reinterpretation. And indeed, there are good arguments in form of novel postulates that are motivated by typical requirements in ontology change settings: One wants to preserve the ontologies somehow in the ontology revision result and also wants them to be reconstructible. In particular these requirements occur when the ontologies are well-developed.

I describe these postulates for the iterated scenario with a sequence SEQ of triggering ontologies. Let $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ be an ontology and let SEQ be a finite sequence of ontology axioms containing only symbols in the public vocabulary $\mathcal{V}$.

An operator $\circ$ that fulfills the iterated preservation postulate for the left argument (Preservation) has to guarantee that there is a substitution σ s.t. the initial ontology O is preserved in internalized form $O\sigma$ in the result of iterated revision with a sequence SEQ.

(Preservation) There is a substitution σ s.t.:

$$O\sigma \subseteq O \circ \mathrm{SEQ}$$

An operator $\circ$ that fulfills (Reconstruction) has to guarantee the existence of a substitution ρ such that the initial ontology O and the set $set(\mathrm{SEQ})$ of all triggering ontologies in the sequence SEQ are contained in a renamed variant of the revision result $(O \circ \mathrm{SEQ})\rho$.

(Reconstruction) There is a substitution ρ s.t.:

$$O \cup set(\text{SEQ}) \subseteq (O \circ \text{SEQ})\rho$$

All reinterpretation operators of this paper fulfill both postulates. I state this proposition the proof of which is a slight adaptation of the proof given in (Eschenbach and Özçep 2010) for triggering concept-based literals.

Proposition 4 *Let $\mathcal{O} = (O, \mathcal{V}, \mathcal{V}')$ be an ontology and* SEQ *be finite sequence of setts of ontology axioms over $\mathcal{V}$ and $\circ$ a reinterpretation operator for triggering ontologies. Then there exists σ and ρ, such that :*

1. $O\sigma \subseteq O \circ \text{SEQ}$
2. $O \cup set(\text{SEQ}) \subseteq (O \circ \text{SEQ})\rho$
3. *For all symbols $C \in \mathcal{V}(O) \cup \mathcal{V}$ one has: $C = C\rho$.*

5 Related Work

The reinterpretation operators are constructed in a similar fashion as those by Delgrande and Schaub (Delgrande and Schaub 2003) but differ in that they are defined not only for propositional logic but also for DLs (and FOLs). Moreover, I consider different stronger forms of bridging axioms than the implications of (Delgrande and Schaub 2003).

Bridging axioms are special mappings that are used in the reinterpretation operator as auxiliary means to implement ontology revision. One may also consider mappings by themselves as the objects of revision (Qi, Ji, and Haase 2009; Meilicke and Stuckenschmidt 2009). A particularly interesting case of mapping revision comes into play with mappings used in the ontology based data access paradigm (Calvanese et al. 2009). These mappings are meant to lift data from relational DBs to the ontology level thereby mapping between close world of data and the open world of ontologies. In this setting different forms of inconsistencies induced by the mappings can be defined (such as local vs. global inconsistency) and based on this mapping evolution ensuring (one form of consistency) be investigated (Lembo et al. 2015).

In this paper I used reinterpretation operators as change operators on ontologies described in DLs. There are different other approaches that use the ideas of belief revision for different forms of ontology change such as ontology evolution over DL-Lite ontologies (Kharlamov, Zheleznyakov, and Calvanese 2013) or ontology debugging (Ribeiro and Wassermann 2009). As the consequence operator over DLs do not fulfill all preconditions assumed by AGM (Alchourrón, Gärdenfors, and Makinson 1985) one cannot directly transfer AGM constructions and ideas one-to-one to the DL setting as noted, e.g., by Flouris and colleagues (Flouris, Plexousakis, and Antoniou 2005) and dealt in more depth for non-classical logics by Ribeiro (Ribeiro 2012). For the definition of the reinterpretation operators the constraint is not essential. Nonetheless, they lead to constraints in providing appropriate counter examples: namely ontologies expressible in the DL at hand.

6 Conclusion

The paper discussed iterative applications of reinterpretation operators meant to handle conflicts due to ambiguous use of symbols in related and well-developed ontologies. Reinterpretation operators may also be used for solving consistencies not due to ambiguity but due to false information—and indeed, the related revision operators in (Delgrande and Schaub 2003) do not talk about ambiguity. But reinterpretation operators cannot be used to solve inconsistencies that clearly cannot be explained by ambiguity: namely consistencies due to different constraints of the sender and the receiver regarding the number of possible objects in the domain (this was discussed under the term *reinterpretation compatibility* in (Özçep 2008)).

The reinterpretation for triggering literals were shown to fulfill (adapted versions of) classical iteration postulates of Darwiche and Pearl (Darwiche and Pearl 1994) whereas the reinterpretation operators for ontologies were shown in general not to fulfill them. Some of the postulates were criticized for general reasons. Nonetheless, still one may consider other forms of reinterpretation operators that incorporate the reinterpretation history in order to define dynamic operators: For example one might weight symbols according to the number of times they were reinterpreted and then use a comparison of the weights in the next iteration step in order to decided which symbols to reinterpret next.

In addition to the general criticisms I discussed the adequateness of the other postulates in view of the special ontology change scenario. Here one cannot guarantee the fulfillment by reinterpretation operators that implement a uniform reinterpretation. But uniformity is necessary in order to guarantee the fulfillment of postulates that express the preservation and reconstructibility of the ontologies in the revision result.

Appendix: Proofs

Proof of Proposition 2

For the proof we need the following lemma

Lemma 1 *For all $X_2 \in \mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1 \cup O_2)$ there is a $X_1 \in \mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1)$ such that $X_2 \subseteq X_1$.*

Proof. If $O\sigma_{\mathcal{V}} \cup O_1 \cup O_2 \cup X_2$ is consistent, so is $O_1 \cup O_2 \cup X_2$. If X_2 is not maximal, then there is a superset X_1 in $\mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1)$. □

Define the following sets

$$\begin{array}{rcl} X_1 & = & \gamma(\mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1)) \\ X_2 & = & \gamma(\mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1 \cup O_2)) \end{array}$$

With this notation we have $(O \otimes^{\gamma} O_1) \cup O_2 = O\sigma_{\mathcal{V}} \cup O_1 \cup O_2 \cup X_1$ and $(O \otimes^{\gamma} O_1) \cup O_2 = O\sigma_{\mathcal{V}} \cup O_1 \cup O_2 \cup X_2$. Assume that $(O \otimes_2^{\gamma} O_1) \cup O_2$ is consistent. Then there is a $X' \in \mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1 \cup O_2)$ with $X_1 \subseteq X'$. Because of Lemma 1 there is a $X'' \in \mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1)$ with $X' \subseteq X''$. Hence $X_1 \subseteq X''$. But as X_1 is inclusion maximal, one gets $X_1 = X'' = X'$ and hence $X_1 \in \mathcal{B}(\sigma_{\mathcal{V}})\top(O\sigma_{\mathcal{V}} \cup O_1 \cup O_2)$. Due to Lemma 1 the first precondition for maximum based selection functions was shown to hold. Now we showed also that the second condition holds, hence $X_1 = X_2$. In particular: $O \otimes^{\gamma} (O_1 \cup O_2) = (O \otimes^{\gamma} O_1) \cup O_2$.

Proof of Proposition 3

All results hold trivially if O is not consistent. So for the following assume that O is consistent.

Proof of 1. In the following let $\circ \in \{\otimes, \boxdot, \oplus^{\mathrm{sel}}\}$.

Proof for (RDP 1): As O_1 and O_2 are trigger literals, $O_2 \models O_1$ holds iff $O_1 = O_2$. Hence $(O \circ O_1) \circ O_2 = (O \circ O_2) \circ O_2 = O \circ O_2$, as the reinterpretation operators fulfill the success postulate.

Counterexample for (RDP 2): Let $O = \{A(b)\}$, $O_1 = \{A(a)\}$ and $O_2 = \{\neg A(a)\}$. Then $O_1 \cup O_2 \models \bot$. On the one hand $O \circ O_2 = \{A(b), \neg A(a)\} \models A(b)$; on the other hand $(O \circ O_1) \circ O_2 = \{A(b), A(a)\} \circ_2 \{\neg A(a)\} \not\models A(b)$ due to Proposition 1.3 (for the weak operators) and Proposition 1.4 (for the strong operators) and again due to 1.3 for the selection based operators $\oplus^{\mathrm{sel}}$, as in this case $(O \otimes O_1) \otimes O_2 \equiv (O \oplus^{\mathrm{sel}} O_1) \oplus^{\mathrm{sel}} O_2$.

Proof for (RDP 3): Let $O \circ O_2 \models O_1$.

Case 1: $O \cup O_2$ is consistent. As O_1 are O_2 literals, $O_2 \models O_1$ means that $O_1 = O_2$. Now $(O \circ O_1) \circ O_2 = (O \circ O_2) \circ O_2 = O \cup O_2 = O \circ O_2$.

Case 2: $O \cup O_2$ is inconsistent. We show the proof for $O_2 = \{A(a)\}$. (The case that $O_2 = \{\neg A(a)\}$ is proved similarly.) Subcase 2.1: $A \notin \mathcal{V}(O_1)$. Then from $O \circ_2 O_2 \models O_1$ and Proposition 1.1, it follows that $O \models O_1$, hence $(O \circ O_1) = O \cup O_1$. Because of 1.1 one has: $(O \cup O_1) \circ O_2 \models O_1$. Subcase 2.2: $A \in \mathcal{V}(O_1)$. Let $\circ = \otimes$. Because of $O \circ O_2 \models O_1$ and Proposition 1.3 it must be the case that O_1 contains A positively, i.e., $O_1 = \{A(c)\}$. Because $\otimes$ fulfills the success postulate $O \otimes \{A(c)\} \models \{A(c)\}$ and hence also $(O \otimes \{A(c)\}) \cup \{a \not\doteq c\} \models \{A(c)\}$. Because of Proposition 1.2 one gets $(O \otimes \{A(c)\}) \otimes_2 \{A(a)\} \models \{A(c)\}$. Now consider the case $\circ = \boxdot$. If A is positive in O_1, then $O_1 = \{A(c)\}$. In this case again Proposition 1.2 gives the result $(O \boxdot O_1) \boxdot O_2 \models O_1$. In the other case O_1 is of the form $O_1 = \{\neg A(c)\}$; using the assumption $O \circ O_2 \models O_1$ one gets with Proposition 1.4 that $O \models \neg\, \mathrm{msc}_O(a)(c)$ and $O \models O_1$. In particular $O \boxdot O_1 = O \cup O_1 \equiv O$. Also $\mathrm{msc}_{O \boxdot O_1}(a) = \mathrm{msc}_O(a)$. Hence from $O \models \neg\, \mathrm{msc}_O(a)(c)$ we get $O \boxdot O_1 \models \neg\, \mathrm{msc}_{O \boxdot O_1}(a)(c)$. Because $O \boxdot O_1 \models O_1$ holds, one can infer with Proposition 1.4 that $(O \boxdot O_1) \boxdot_2 O_2 \models O_1$. With a similar argument and Proposition 1.5 one can show that the results holds for $\circ = \oplus^{\mathrm{sel}}$.

Proof for (RDP 4): Let $O_1 \cup (O \circ O_2)$ be consistent.

Case 1: $O \cup O_2$ is consistent. Then $O \circ O_2 = O \cup O_2$ and so $O_1 \cup (O \cup O_2) \not\models \bot$ due to assumption. But then $O \circ O_1 = O \cup O_1$.

Case 2: $O \cup O_2$ is not consistent. If also $(O \circ O_1) \cup O_2$is consistent, then $(O \circ O_1) \circ O_2 = (O \circ O_1) \cup O_2$. As $O \circ O_1 \models O_1$ we then have $(O \circ O_1) \circ O_2 \models O_1$, in particular $O_1 \cup (O \circ O_1) \circ O_2) \not\models \bot$. Therefore consider now the case that $(O \circ O_1) \cup O_2$ is inconsistent.

We show the result for $O_2 = \{A(a)\}$ (The argument for negative literals is similar). Subcase 2.1: $A \notin \mathcal{V}(O_1)$. Due to success, $O \circ O_1 \models O_1$ and with Proposition 1.1 it follows that $(O \circ O_1) \circ O_2 \models O_1$. Because $(O \circ O_1) \circ O_2$ is consistent so is $O_1 \cup (O \circ O_1) \circ O_2$. Subcase 2.2: $A \in \mathcal{V}(O_1)$. Assume $O_1 \cup (O \circ O_1) \circ O_2 \models \bot$. Is O_1 of form $O_1 = \{\neg A(c)\}$, then due to Proposition 1.2 it holds that $O_1 \cup O \circ O_1 \cup \{a \not\doteq c\}$ is not consistent. Because $O \circ O_1 \models O_1$ this can be the case only if $O \circ O_1 \models a \doteq c$. Then $O_1 = \{\neg A(a)\}$ which contradicts the assumption $O_1 \cup (O \circ O_2) \not\models \bot$.

If O_1 has the form $O_1 = \{A(c)\}$, then, due to Proposition 1.3, this can only be the case if $\circ = \boxdot$ or $\circ = \oplus^{\mathrm{sel}}$. With the assumption that $(O_1 \cup (O \circ O_1) \circ O_2$ is not consistent, i.e. $(O \circ A(c)) \circ A(a) \models \neg A(c)$, one could infer with Proposition 1.4 and 1.5 that $O \circ A(c) \models \neg A(c)$ which would mean that $O_1 \cup (O \circ O_1)$ is not consistent—-contradicting the consistency of $O \boxdot O_1$ and the fact that $O \boxdot_2 O_1 \models O_1$.

Proof of 2. Counter example for (RDP 1): Consider

$$\begin{array}{rcl} O & = & \{\neg A(a)\} \\ O_1 & = & \{(A \sqcup B)(a)\} \\ O_2 & = & \{A(a)\} \end{array}$$

Let be γ an arbitrary selection function that prefers the reinterpretation of role and concept symbols. Then we get on the one hand $O \otimes^\gamma O_2 = \{\neg A'(a'), A(a), A' \sqsubseteq A, a \doteq a'\}$. And so $O \otimes^\gamma O_2 \not\models B(a)$. On the other hand $O \otimes^\gamma O_1 = \{\neg A(a), (A \sqcup B)(a)\} \models B(a)$. And last

$$\begin{array}{rcl} (O \otimes^\gamma O_1) \otimes^\gamma O_2 & = & \{\neg A'(a'), (A' \sqcup B')(a'), A(a), \\ & & A' \sqsubseteq A, a \doteq a', B \sqsubseteq B', \\ & & B' \sqsubseteq B\} \end{array}$$

But then $(O \otimes^\gamma O_1) \models B(a)$,

Counter example for (RDP 2): See counter example for triggering literals.

Counter example for (RDP 3):

$$\begin{array}{rcl} O & = & \{A(a), \exists R_1.A \sqsubseteq \neg B, R_1(a,c), \exists R_2.A \sqsubseteq A, \\ & & R_2(b,e)\} \\ O_1 & = & \{\neg A(b)\} \\ O_2 & = & \{\neg A(a), B(a), A(e), \exists R_3.A \sqsubseteq A, R_3(c,b)\} \end{array}$$

For all selection functions γ that prefer the reinterpretation of concept and role symbols one has:

$$\begin{array}{rcl} O \otimes^\gamma O_1 & = & O \cup O_1 \\ O \otimes^\gamma O_2 & = & O\sigma_{\mathcal{V}} \cup O_2 \cup \mathcal{B}(\sigma_{\mathcal{V}}) \setminus \{A' \sqsubseteq A\} \\ & \models & \neg A(b) \\ (O \otimes^\gamma O_1) \otimes^\gamma O_2 & = & (O \cup O_1)\sigma_{\mathcal{V}} \cup O_2 \cup \\ & & \mathcal{B}(\sigma_{\mathcal{V}}) \setminus \{A' \sqsubseteq A, A \sqsubseteq A'\} \\ & \not\models & \neg A(b) \end{array}$$

Note that the O, O_1, O_2 have simple structures and do not presuppose complex DL constructors.

Counter example for (RDP 4): Consider:

$$\begin{array}{rcl} O & = & \{B(a), B(b) \vee C(b)\} \\ O_1 & = & \{\neg A(a), \neg B(b)\} \\ O_2 & = & \{\neg B(a) \vee A(a), \neg B(b), \neg C(b)\} \end{array}$$

Choose γ such that the following results hold:

$$\begin{array}{rcl} O \otimes_2^\gamma O_2 & = & \{B'(a'), B'(b') \vee C'(b'), \\ & & \neg B(a) \vee A(a), \neg B(b), \neg C(b), \\ & & a \doteq a', b \doteq b', \\ & & C \sqsubseteq C', C' \sqsubseteq C, B \sqsubseteq B'\} \\ & \not\models & \neg \bigwedge O_1 \end{array}$$

$$\begin{aligned} O \otimes_2^\gamma O_1 &= \{B(a), B(b) \vee C(b), \\ &\quad \neg A(a), \neg B(b)\} \models C(b) \\ (O \otimes_2^\gamma O_1) \otimes_2^\gamma O_2 &= \{B'(a'), B'(b') \vee C'(b'), \\ &\quad \neg A'(a'), \neg B'(b'), \neg B \sqcup A(a), \\ &\quad \neg B(b), \neg C(b), a \doteq a', b \doteq b', \\ &\quad B \sqsubseteq B', B' \sqsubseteq B, C \sqsubseteq C', \\ &\quad A' \sqsubseteq A\} \models \neg \bigwedge O_1 \end{aligned}$$

Note that we use here boolean ABoxes. As in the previous counterexample a purely DL counter example with standard ABoxes should be constructible.

Proof of 3. Counter examples for (RDP 1), (RDP 3) und (RDP 4): Consider the same set of ontology axioms as in the corresponding counter examples for the weak reinterpretation $\otimes^\gamma$. The selection function γ can be chosen such that the same results follow as in the case of $\otimes^\gamma$.

Counter example for (RDP 2): See counter example for triggering literal.

References

Ahmeti, A.; Calvanese, D.; and Polleres, A. 2014. Updating RDFS aboxes and tboxes in SPARQL. In Mika, P.; Tudorache, T.; Bernstein, A.; Welty, C.; Knoblock, C. A.; Vrandecic, D.; Groth, P. T.; Noy, N. F.; Janowicz, K.; and Goble, C. A., eds., *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, volume 8796 of *Lecture Notes in Computer Science*, 441–456. Springer.

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change: partial meet contraction and revision functions. *Journal of Symbolic Logic* 50:510–530.

Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; Poggi, A.; Rodríguez-Muro, M.; and Rosati, R. 2009. Ontologies and databases: The DL-Lite approach. In Tessaris, S., and Franconi, E., eds., *Semantic Technologies for Informations Systems – 5th Int. Reasoning Web Summer School (RW 2009)*, volume 5689 of *Lecture Notes in Computer Science*. Springer. 255–356.

Darwiche, A., and Pearl, J. 1994. On the logic of iterated belief revision. In Fagin, R., ed., *Proceedings of the 5th Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-94)*, 5–23.

Darwiche, A., and Pearl, J. 1997. On the logic of iterated belief revision. *Artificial intelligence* 89:1–29.

Delgrande, J. P., and Schaub, T. 2003. A consistency-based approach for belief change. *Artificial Intelligence* 151(1–2):1–41.

Delgrande, J. P. 2008. Horn clause belief change: Contraction functions. In Brewka, G., and Lang, J., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the 11th International Conference, KR 2008, Sydney, Australia, September 16-19, 2008*, 156–165. AAAI Press.

Eschenbach, C., and Özçep, Ö. L. 2010. Ontology revision based on reinterpretation. *Logic Journal of the IGPL* 18(4):579–616. First published online August 12, 2009.

Flouris, G.; Manakanatas, D.; Kondylakis, H.; Plexousakis, D.; and Antoniou, G. 2008. Ontology change: classification and survey. *The Knowledge Engineering Review* 23(2):117–152.

Flouris, G.; Plexousakis, D.; and Antoniou, G. 2005. On applying the AGM theory to dls and OWL. In Gil, Y.; Motta, E.; Benjamins, V. R.; and Musen, M. A., eds., *The Semantic Web - ISWC 2005, 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005, Proceedings*, volume 3729 of *Lecture Notes in Computer Science*, 216–231. Springer.

Freund, M., and Lehmann, D. J. 2002. Belief revision and rational inference. *Computing Research Repository (CoRR)* cs.AI/0204032.

Goeb, M.; Reiss, P.; Schiemann, B.; and Schreiber, U. 2007. Dynamic TBox-handling in agent-agent-communication. In Beierle, C., and Kern-Isberner, G., eds., *Dynamics of Knowledge and Belief. Proceedings of the Workshop at the 30th Annual German Conference on Artificial Intelligence (KI-2007)*, 100–117. Fernuniversität in Hagen.

Gutierrez, C.; Hurtado, C.; and Vaisman, A. 2011. Rdfs update: From theory to practice. In *Proceedings of the 8th Extended Semantic Web Conference on The Semanic Web: Research and Applications - Volume Part II*, ESWC'11, 93–107. Berlin, Heidelberg: Springer-Verlag.

Kharlamov, E.; Zheleznyakov, D.; and Calvanese, D. 2013. Capturing model-based ontology evolution at the instance level: The case of dl-lite. *J. Comput. Syst. Sci.* 79(6):835–872.

Lembo, D.; Mora, J.; Rosati, R.; Savo, D. F.; and Thorstensen, E. 2015. Mapping analysis in ontology-based data access: Algorithms and complexity. In et al., M. A., ed., *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I*, volume 9366 of *Lecture Notes in Computer Science*, 217–234. Springer.

Meilicke, C., and Stuckenschmidt, H. 2009. Reasoning support for mapping revision. *Journal of Logic and Computation.*

Özçep, Ö. L. 2008. Towards principles for ontology integration. In Eschenbach, C., and Grüninger, M., eds., *FOIS*, volume 183, 137–150. IOS Press.

Özçep, Ö. L. 2012. Minimality postulates for semantic integration. In Konieczny, S., and Meyer, T., eds., *Proceedings of the workshop BNC@ECAI2012*, 47–53.

Qi, G.; Ji, Q.; and Haase, P. 2009. A conflict-based operator for mapping revision. In et al., B. C. G., ed., *Proceedings of the 22nd International Workshop on Description Logics (DL-09)*, volume 477 of *CEUR Workshop Proceedings.*

Ribeiro, M. M., and Wassermann, R. 2009. Base revision for ontology debugging. *J. Log. Comput.* 19(5):721–743.

Ribeiro, M. 2012. *Belief Revision in Non-Classical Logics*. SpringerBriefs in Computer Science. Springer.

Law Tests for Semantically-Safe Rule Interchange

Adrian Paschke and Tara Athan
Freie Universitaet Berlin

Abstract

In this paper, we extend a highly successful trend in software engineering (SE), namely test-driven development, to define self-validating rule bases to safeguard rule interchange in distributed environments such as the Web. The concept of a law test is introduced as a way of compactly representing a collection of test matching a predefined pattern, or "law". Sets of law tests, associated with multiple laws, can be used to provide evidence of the semantically correct execution of an interchanged rule-based Logic Program (LP) in a target execution environment, such as a inference service or agent, by running, in the target inference engine, the law tests attached to and interchanged with the LP. Even in cases when the service/agent does not provide explicit information about its supported semantics, e.g. by its interface description, some failed law tests can be used to prove that the intended semantic properties for the interchange rule program are violated by the implemented semantics, while successful law tests can provide evidence that the intended semantics is supported.

1 Introduction

A strong demand for rule-based functionalities comes from the distributed systems and Web community, in particular in the area of Semantic Web (SW). Here ever larger and more complex rule bases are increasingly managed and maintained in a distributed environment and interchanged over domain and system boundaries between platform-specific inference engines using more-or-less standardized rule markup interchange formats such as RuleML[1]. The correct execution of the interchanged rule-based LPs depends on both the intended semantics of the interchanged LP and the actual (implemented) semantics of the inference engine used at the target SW service/agent.

The contribution of this paper is a test-driven approach supporting semantically-safe rule interchange and verification & validation (V&V) of inference engines provided e.g. as open SW inference services or rule-based agents on the Web. Tests, which are typically interchanged together with an LP, can support decisions as to whether that LP has the intended behavior if it is executed in the target environment (Paschke 2006; Paschke et al. 2006; Dietrich and Paschke 2005). Our new contribution are "law tests", which probe the supported laws of the logic of a knowledge representation (KR) language of interest and their semantic properties (LP semantics and non-monotonic semantics).

We first present some background in LP semantics in section 2. In section 3 we introduce our conceptual solution using law tests for probing the laws of logic of an inference engine. In section 4 we further refine this idea of law tests for deciding if an interchange rule program can be correctly interpreted with the semantics supported by the target inference service/agent. We demonstrate and evaluate the concept by a proof-of-concept implementation in section 5. In section 6 we discuss related work and we conclude the paper in section 7 with future work.

2 Background

It is beyond the scope of this paper to review or compare all the different LP- and NMR (non-monotonic reasoning)-semantics. For an overview we refer to (Apt and Bol 1994; Minker 1993; Dix 1995a). We give some general definitions and an initial taxonomy of semantics and LP classes.

Definitions

For the scope of this paper, we will assume all intended and implemented semantics are based on the 3-valued Herbrand semantics of the connectives $\wedge$, $\vee$, $\neg$ as well as "weak" (Lukasiewicz) implication $\leftarrow$ and equivalence $\leftrightarrow$.

- A semantics SEM of a class of programs $\mathcal{L}$ (the LP class of SEM) is a mapping that assigns to each member P of $\mathcal{L}$ a value SEM_P in the powerset of 3-valued Herbrand models on $\mathcal{L}_{\mathcal{P}}$, the restriction of $\mathcal{L}$ to the nonlogical constants that appear in P.
- If U is a set of atoms, then $SEM_{P \cup U}$ may be written as $SEM_P(U)$.
- A sceptical entailment relation ($SEM_{scept,P}$ or $\mid\!\sim_P$) is defined from sets of atoms to sets of literals, such that $U \mid\!\sim_P V$ iff every $v \in V$ is a Lukasiewicz consequence of M ($M \models V$) for every $M \in SEM_P(U)$. If U is the empty set, it may be omitted. The entailment relation is extended to single atoms and literals by considering them as singleton sets.
- A semantics SEM' *extends* a semantics SEM (denoted by $SEM' \geq SEM$) iff for all P in $\mathcal{L}_{\mathcal{P}}$, the LP class

[1] http://ruleml.org

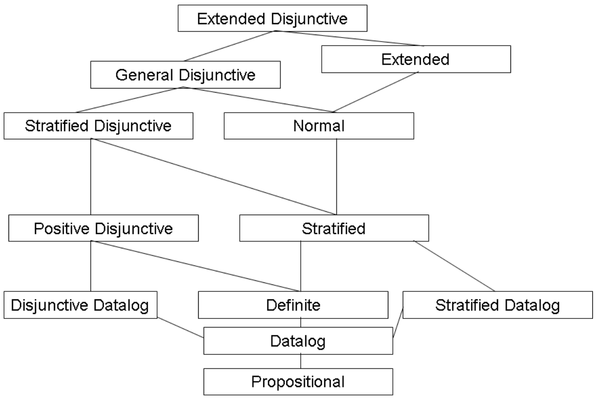


Figure 1: Classes of LPs. Lines between classes represent containment, with the larger (more expressive) class higher in the diagram.

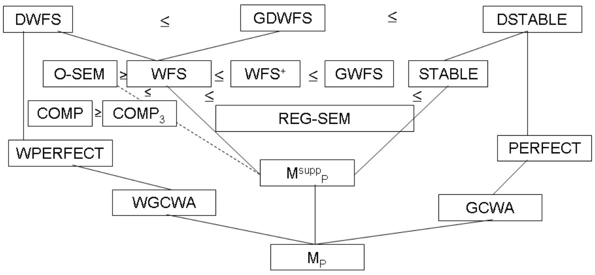


Figure 2: Semantics for LPs and their Extension Relations.

of SEM, $SEM'_{scept,P} \supseteq SEM_{scept,P}$. Note that if SEM' extends SEM, then the LP class of SEM' $\mathcal{L}'_{\mathcal{P}} \supseteq \mathcal{L}_{\mathcal{P}}$.

- A semantics SEM' *conservatively extends* a semantics SEM (denoted by $SEM' \geq_c SEM$) iff SEM' extends SEM and $SEM'_{scept,P}$ equals $SEM_{scept,P}$ for all $P \in \mathcal{L}_{\mathcal{P}}$, the LP class of SEM.
- Lukasiewicz tautologies, e.g. $a = p \leftarrow p$, are denoted $\models a$.

Figure 1 shows different classes of LPs partially-ordered by expressiveness, i.e. a more expressive class Q' includes a less expressive class Q. For example, the class of extended disjunctive LPs is more expressive than the class of normal LPs; it extends the normal LP class with explicit negation and disjunctions.

Figure 2 illustrates the partial orderings $\geq$ and $\geq_c$ of semantics on LP classes up to general disjunctive LPs, with semantics having larger LP classes higher in the diagram. Lines between the semantics represent conservative extensions ($\geq_c$), while non-conservative extensions are indicated with $\geq$.

An LP P can be executed by an inference engine if and only if all of the following holds:

- P is in the LP class of the implemented semantics SEM' of the inference engine.
- the literals entailed by P under the implemented semantics of the inference engine that are in the LP class of the intended semantics SEM are entailed under the intended semantics.
- the literals entailed by P under the intended semantics SEM are entailed under the implemented semantics of the inference engine.

If $SEM' \geq_c SEM$, then all conditions above hold; thus it is a sufficient (but not necessary) condition for P to be executable by an inference engine that the implemented semantics be a conservative extension of the intended semantics[2]. This information about the semantics may be given by explicit annotations, e.g. referencing predefined semantic profiles (see e.g. RuleML's semantic profile mechanism (Paschke 2014)), or it may need to be determined.

3 Concept - Law Tests

In the context of rule interchange with open, distributed inference engines, which might be provided as open inference services, an important question is, whether the inference engine correctly implements a semantics. Tests, possibly interchanged together with the LP, can be used to support verification and validation of the execution of the interchanged LP in the target environment and therefore establish trust in this service.

Following sofware engineering standards, e.g. (ISO/IEC/IEEE 29119-1 2013), we make use of the following terminology about testing in general:

- a *test case* is a performative, typically a query, together with the expected result.
- A *test suite*, a set of assertions (which may be logic programs or ground facts) together with a set of tests.
- A *test* is a test case or a test suite.
- A *test program* is a logic program asserted within a test suite.

At the lowest level of evaluation, the performative of each test case is executed on the appropriate assertions (if any). It succeeds if the actual results agree with the expected results, and fails otherwise, i.e., if there is no termination or the actual results do not agree with the expected results. The typical *test protocol* for a test suite is that it succeeds if and only if all the tests it contains succeed, but other test protocols are allowed, as we will see below.

For example, consider a test suite asserting a simple test program with the following rules:

Example 3.1. Simple LP

```
a <- ¬b
b <- ¬a
c <- a
c <- b
```

[2] However note, that the complexity of SEM' in most cases is higher then in SEM if SEM' is an extension of SEM, and hence for performance reasons an inference engine supporting only the intended semantics SEM is typically optimal for programs of the LP class for which SEM is defined.

and containing one test case $T = \{c? \ => \ true\}$, i.e., the test query performative $c?$ which as expected result should yield true. In the case that the inference engine supports stable model semantics (STABLE) this test case, and the test suite, succeeds.

However, if the inference engine supports well-founded semantics (WFS) the test query leads to the actual result ”unknown” and accordingly the test case, and the test suite, fails.

While the test suite described above directly relates to the V&V of a particular interchanged LP with a particular semantics, it is a cumbersome task to develop such tests. Further, these tests are limited in scope - in a context where one of a limited number of semantics is known to be correctly implemented, one or a few such tests may be sufficient to discriminate between the possible implemented semantics. However, a systematic approach to developing tests based on the properties, or *laws*, of the intended semantics is needed to support target engine selection in a more general context.

We define a *law test* to be a test whose failure indicates that a particular law does not hold in the implemented semantics, and whose success provides evidence that the law holds in the implemented semantics. In general, a *law test suite* is a test suite that is a law test, and a *law test case* is a law test that is a test case. In this paper, we are concerned only with a particular form of law test suite where the tests it contains are organized into two suites: a pre-test suite and a post-test suite with one test case. For the law test to succeed, either the pre-test suite must fail or the post-test suite must succeed.

Law tests can be used to assess the correctness of the semantics of an inference engine or to automatically determine the semantics of an inference engine, e.g., in cases when the semantics of the receiving inference service/agent is not explicitly stated. Individual law tests are used to probe the laws of an implemented semantics, such as the supported rules of inference, or its structural properties. By applying a number of law tests to a particular inference service/agent, the implemented semantics provided by the engine can be partially characterized by the set of successful and failed law tests.

For example, SLDNF resolution typically suffers from loops, whereas e.g. SLG resolution implements some form of loop checking and loop prevention. A simple law test for (partially) verifying the “law” of loop avoidance might be, e.g., the following test program:

Example 3.2. Loop LP

```
a
b
a <- b
b <- a
```

and the test case $\{a? \ => \ true\}$, which succeeds for an inference engine supporting reflexivity and loop prevention, e.g. WFS, whereas for an inference engine supporting COMP and SLDNF resolution it does not terminate, demonstrating the lack.

Another problem of SLDNF resolution is that it cannot handle free variables in negative subgoals due to the procedural negation-as-finite-failure, e.g.:

Example 3.3. Free Variables in Negative Subgoals LP

```
q(b)
r(f(a))
p(X) <- ¬q(X), r(f(X))
```

The test case $\{p(a)? \ => \ true\}$ succeeds when the implemented semantics is SLDNF. However, a test case $\{p(X)? \ => \ true \ : \ \{X/a\}\}$ with the expected variable binding $X = a$ fails, because the negation-as-failure tree is entered with a free variable, which fails due to the fact that no variable binding for X is computed, even though $r(f(X))$ holds and $q(a)$ does not. This problem is also known as the floundering problem, so the above test may be considered a law test for the “no-floundering law”.

For more unsolvable problems that can be used for testing SLDNF semantics see e.g. (Shepherdson 1991). Other well-known problems and paradoxes from literature, e.g. Yale Shooting Problem, which are solved by a particular semantics or resolution, resp., can also be used as tests.

Tests of this sort can be bundled into larger suites of law tests for assessing the validity of certain inference engine implementations (resolution algorithm), semantics (e.g. by testing the entailments) and logical calculi. These suites of law tests can be either interchanged directly together with a rule program or provided in a public testing repository accessible via a service interface, e.g., provided by standards body such as RuleML.

4 Solution - Semantically Safe Rule Interchange with Law Tests

To systematically test if an interchanged rule program (LP) can be correctly interpreted with the supported semantics of an inference service/agent we have proposed above the use of suites of law tests for the intended semantic properties of the interchanged program. In this section, we consider how this approach can be applied to some known laws that effectively discriminate between existing LP semantics.

Kraus et al. (Kraus, Lehmann, and Magidor 1990) and Dix (Dix 1995b; 1995c) proposed several weak and structural (strong) properties for arbitrary (non-monotonic) semantics. In the following we will briefly review these properties and show how they can be adapted to law tests.

The following structural properties for a sceptical entailment relation $\mid\!\sim_P$ are adapted from similar properties for a classical logic language L as stated in (Kraus, Lehmann, and Magidor 1990) and similar properties for normal LPs in (Dix 1995b), where here the conventional proof format is used for meta-level implications, the subscript is dropped on the sceptical entailment relation when not needed, a, b are atoms, f, g are literals, and C is a set of atoms, J, K are sets of literals, P, Q are programs :

- *Right Weakening*: $\frac{\models g \leftarrow f, \ C \mid\!\sim f}{C \mid\!\sim g}$
- *Reflexivity*: $C \mid\!\sim C$
- *And*: $\frac{C \mid\!\sim J, \ C \mid\!\sim K}{C \mid\!\sim J \cup K}$
- *Or*: $\frac{a \mid\!\sim_P J, \ b \mid\!\sim_P J}{\mid\!\sim_{P \cup \{a \vee b\}} J}$
- *Left Logical Equivalence*: $\frac{\models f \leftrightarrow g, \ f \mid\!\sim J}{g \mid\!\sim J}$

- *Cautious Monotony*: $\frac{\mid\!\sim_P J,\ \mid\!\sim_P K}{\mid\!\sim_{P\cup J} K}$
- *Cut*: $\frac{\mid\!\sim_P J,\ \mid\!\sim_{P\cup J} K}{\mid\!\sim_P K}$
- *Cumulativity (Cut and Cautious Monotony)* If $\mid\!\sim_P J$ then $\mid\!\sim_{P\cup J} K$ if and only if $\mid\!\sim_P K$
- *Rational Monotony*: $\frac{\mid\!\not\sim_P \neg f,\ \mid\!\sim_P J}{\mid\!\sim_{P\cup f} J}$
- *Disjunctive Rationality*: $\frac{a \mid\!\not\sim_P J,\ b \mid\!\not\sim_P J}{\mid\!\not\sim_{P\cup\{a\vee b\}} J}$
- *Negation Rationality*: $\frac{\mid\!\not\sim_{P\cup a} J,\ \mid\!\not\sim_{P\cup\neg a} J}{\mid\!\not\sim_P J}$

In addition to these structural/strong properties the following weak properties describing general conditions a reasonable (sceptical) semantics should satisfy are derived from (Dix 1995c) (programs consist of a set of rules of the form $H \leftarrow B$ where H and B are sets of formulas and a set of facts that are formulas):

- *Elimination of Tautologies*: If a tautological rule ($\models H \leftarrow B$) is eliminated from a program P, then the resulting program P' is semantically equivalent. I.e., $SEM_{scept,P} = SEM_{scept,P'}$ where $P' = P \setminus \{H \leftarrow B\}$ and $H \subseteq B$.
- *Generalized Principle of Partial Evaluation (GPPE)*: If a rule $H \leftarrow B$, where B contains a subgoal b, is replaced in a program P' by the n rules $H \cup (H^i - b) \leftarrow ((B - b) \cup B^i)$, where $H^i \leftarrow B^i$ are all rules for which $b \in H^i$, then the $SEM_{scept,P} = SEM_{scept,P'}$
- *Positive/Negative Reduction*: If a rule $H \leftarrow B$ is replaced in a program P' by $H \leftarrow B \backslash \{\neg c\})$ (c is a formula) where c appears in no rule head, or a rule $H \leftarrow B$ is deleted from P where there is a fact a in P such that $\neg a \subseteq B$, then $SEM_{scept,P} = SEM_{scept,P'}$
- *Elimination of Non-Minimal Rules/Subsumption*: If a rule is deleted from a program P when there is another rule that subsumes it, then the resulting program P' is semantically equivalent. I.e., $SEM_{scept,P} = SEM_{scept,P'}$ if there are distinct rules $H \leftarrow B$ and $H' \leftarrow B' \in P$ such that $B \subset B'$ and $H \subset H'$ and $P' = P \setminus \{H \leftarrow B\}$
- *Closure*: The reduction of P by its entailments has no new entailments. I.e., $SEM_{scept,P^{SEM_{scept,P}}} = \emptyset$, where the reduction P^M of a program by a consistent set of literals M is defined in (Dix 1995c) as a transformation from $\mathcal{L}_P$ into $\mathcal{L}_{P\backslash M}$.
- *Independence*:The truth value of a literal c with respect to a semantics SEM_P does not change when P is extended by a program P' in a disjoint language. I.e., $\mid\!\sim_P c$ iff $\mid\!\sim_{P\cup P'} c$ provided that the languages of P and P' are disjoint and c belongs to the language of P.
- *Relevance*: The truth value of a literal c with respect to a semantics SEM_P, only depends on the subprogram formed from the *relevant rules* of P ($relevant(P)$) with respect to c: $\mid\!\sim_P c$ iff $\mid\!\sim_{relevant(P,c)} c$

Table 1 shows for some common semantics the properties that they satisfy.

In the first category of properties for a reasonable semantics we mainly focus on *Rational Monotony* and *Cumulativity*, since in the settings of the redefined sceptical consequence relation the properties *Right Weakening, Reflexivity, Left Logical Equivalence* are trivial and always satisfied for the intended sceptical semantics[3] which we consider in this paper. *Cumulativity* can be tested using tests for *Cut* and *Cautious Monotony*, since it is equivalent to their combination. Further, satisifaction of *Rational Monotony* implies *Cautious Monotony*, *Disjunctive Rationality* and *Negation Rationality*: $Rationality \Rightarrow DisjunctiveRationality \Rightarrow Negation\ Rationality$.

The other properties such as *Or* are useful in the context of disjunctive LPs, e.g. to distinguish between an exclusive and an inclusive interpretation of $\vee$. *Cut* is a natural condition for non-monotonic formalisms. To test *Cut* resp. *Cautious Monotony* we only need to add those atoms that have to be added in order to satisfy *Cut* resp. *Cautious Monotony* such that $SEM_P \leq SEM_P(M_i)$ where M_i is as sequence of atoms a added to P with $M_0 = \emptyset$. *Rational Monotony* is in any sceptical semantics a stronger form of *Cautious Monotony* because $a \mid\!\sim b \Rightarrow not\ a \mid\!\sim \neg b$. *Rational Monotony* can be inductively proved $SEM_P(M_i) \leq SEM_P(M_{i+1})$ with $M_0 = \emptyset$ and a maximum M_j.

If we can find an extension of atoms to an initial program such that this extension violates the property under test, we have a counter example, and the law test which queries for this atom will have an expected result of false. We reuse this counter example as a law test case to assess whether an arbitrary inference engine, i.e., the semantics implemented by this inference engine, solves this counter example. If the law test is successful the inference engine may satisfy this property. To demonstrate this approach we will now give some examples for testing the properties as defined by (Dix 1995b; 1995c).

Given LPs P_0, P_1, ... and P' as well as multiple test cases, T_0, T_1, ... and T', the law test suite $\{(P_0, T_0), (P_1, T_1), ...\}$, called the "pre-test suite" and (P', T'), called the "post-test", together form a conditional law test. The conditional law test fails if every test case T_i in the pre-test suite is passed when applied to its test program P_i and the post-test case T' fails when applied to its test program P' – otherwise, the conditional law test succeeds.

An engine might fail the pre-test suite due to some P_i being outside of its input language, or when the expected results are not obtained because preconditions of the semantic property are not satisfied. In either of these cases, it is not possible to draw a conclusion about the property, whether the post-test fails or not - this is not evidence that the law holds. However, each such success of a law test when the pre-test suite succeeds provides some evidence that the law holds for the implemented semantics, and the accumulation of the evidence from multiple law tests may be used to inform a decision about the applicability of the implemented engine.

Example 4.1. *STABLE does not satisfy Cautious Monotony*

```
P0: a <- ¬b          P': a <- ¬b
```

[3] Such laws are still useful in V&V of implemented semantics.

Table 1: Table (General Properties of Semantics)

Semantics	Class	Cumul.	Rat.	Taut.	GPPE	Red.	Non-Min.	Rel.	Cons.	Indep.
COMP	Normal	-	●	-	●	●	●	-	-	-
$COMP_3$	Normal	●	●	-	●	●	●	-	-	-
WFS	Normal	●	●	●	●	●	●	●	●	●
STABLE	Normal	-	●	●	●	●	●	-	-	-
WGCWA	Pos. Disj.	-	●	-	●	●	-	●	●	●
CGWA	Strat. Disj.	●	-	●	●	●	●	●	●	●
PERFECT	Strat.Disj.	●	-	●	●	●	●	-	●	●

```
    b <- ¬a                b <- ¬a
    c <- ¬c                c <- ¬c
    c <- a                c <- a
                          c
T0=T′: a=>true, c=>true
```

$STABLE_P$ has $\{a, \neg b, c\}$ as its only stable model and hence it derives a and c, i.e. $T0$ succeeds. But, by adding the derived atom c to P we get $P' := P0 \cup \{c\}$ where $STABLE_{P'}$ has another stable model $\{\neg a, b, c\}$, i.e. a can no longer be derived (i.e. T' fails) and cautious monotony is not satisfied.

Example 4.2. *O-SEM is not rational*

```
P:  a <- ¬a                P′:   a <- ¬a
    p <- a                       p <- a
    q <- ¬p                      q <- ¬p
                                 a
T0 = T′  : ¬p => true, q=>true
```

$O-SEM_P$ derives $\{\neg p, q\}$ from P since $\neg a$ is not derivable. Hence, $T0$ succeeds. But $P' := P \cup a$ derives $\{a, p\}$ and therefore $\neg p$ and q are not derivable (T' now fails), as *Rational Monotony* would require.

Example 4.3. *EWFS does not satisfy CUT*

```
P:  d <- ¬c             P′: d <- ¬c
    a <- ¬a                 a <- ¬a
    b <- ¬x, a              b <- ¬x, a
    c <- ¬b                 c <- ¬b
                             b
T0 = T′:= a=>true, b=>true, ¬c=>fail, d=>fail
```

$EWFS_P$ entails $\{a, b, \neg x\}$ and hence $\neg c$ and d fail, i.e. $T0$ succeeds. But $EWFS_P(\{b\})$ entails $\{a, b, \neg x, \neg c, d\}$. Thus T' fails for the extended program P' with the derivable atom b added; thus *Cut*, and accordingly *Cumulativity* are not satisfied.

Similarly, conditional law test to show that REG-SEM is not Cumulative may be constructed.

Note that in all conditional law test examples above, the post-test consists of a program that is an extension of the program of the pre-test, and the test cases for pre- and post-test are identical. The laws being considered here (Cumulativity, Cautious Monotony, and Rational Monotony) are particularly amenable to conditional law tests of this compact form; however, for other laws, and for broader testing, e.g. for V&V, more general conditional law tests may be considered.

We will now take a look at the second kind of properties for a reasonable semantics, the weak properties introduced above. These properties are defined with respect to a semantic equivalence relation $SEM_P = SEM_{P'}$ for a particular kind of program transformations from P to P'. In a similar way as with the first kind of strong properties we can provide sets of "pre-tests" and "post-tests", which together constitute a conditional law test. The post-tests are derived from the initial programs via applying the defined transformations of the respective property/law of concern. Hence, if the pre-test succeeds and the post-test fails it proves that the implemented semantics does not satisfy this property.

Again, we will illustrate this with some examples:

Example 4.4. *STABLE does not satisfy Relevance*

```
P0: a <- ¬b             P′: a <- ¬b
                            c <- ¬c
T0=T′: a=>true
```

The unique stable model of $P0$ is $\{a\}$. If the rule $c \leftarrow \neg c$ is added, a is no longer derivable because no stable model exists. *Relevance* is violated, because the truth value of a depends on atoms that are totally unrelated to a.

Example 4.5. *GWFS does not satisfy GPPE*

```
P0: d <-¬b             P′: d <-¬b
    a <-¬b                 a <-¬b
    b <- c                 b <- d,¬a
    c <- d,¬a              c <- d,¬a
T0=T′: ¬c=>true, ¬b=>true, d=>true, a=>true
```

The two-valued Herbrand models of $P0$ are $\{a, d\}$ and $\{b\}$. Since c is in neither minimal model, then $\neg c$ is derived by negation-as-failure (naf) in GWFS. In the next iteration, the rule with c as subgoal is removed, so $\neg b$ is derived by negation-as-failure. Finally, $\{a, d\}$ are derived based on $\neg b$. Thus $P0$ entails $\{a, \neg b, \neg c, d\}$ However, for P', $\neg b$ does not follow by naf, since c does not occur in the body of the rule having b in the head; thus, P' entails only $\{\neg c\}$. Hence although P' partially evaluates $P0$, they are semantically not equivalent, which violates the principle of *GPPE*.

Such law tests for checking for violations of either strong or weak semantic properties provides us with a tool for supporting the decision of what semantics should be used for a particular application. For example, an application might require small rule bases, e.g. because the rules are interchanged frequently. Obviously, here the principles of *Elimination of Tautologies* and *Elimination of Non-Minimal Rules* are important, in order to keep the rule base as small as possible without changing its semantics.

Moreover, by taking both kinds of properties together a semantics might be characterized by these, i.e. via applying the complete law test suite consisting of the initial pre-test programs and the post-tests, we can gather evidence as to which properties are satisfied by an arbitrary semantics and which are not. The initial pre-tests not only check the conditions of the semantic properties, they also help to determine whether this particular law test can be applied at all, e.g. it might already give us a clue about the LP class supported by the inference engine of concern. For example, a test program including disjunctions or explicit negation might not be supported by the inference engine.

Given that the pre-tests succeed, the failure of the post-test provides a counter-example of the particular property of concern. The derived set of satisfied and unsatisfied properties for a particular unknown semantics can then be compared to known results for such properties of different semantics. If the set of satisfied and unsatisfied properties derived by applying the meta test programs in the target inference engine, matches the satisfied and unsatisfied properties of a semantics for a particular LP class, this provides evidence that the inference engine supports/implements this semantics.

The semantic principles described in this section are also very important in the context of applying refactoring to LPs. In general, a refactoring to a rule base should optimize the rule code without changing the semantics of the program. Removing tautologies or non-minimal rules or applying positive/negative reductions are typically applied in rule base refinements using refactoring (Dietrich and Paschke 2005) and the semantic equivalence relation between the original and the refined program defined for these principles is therefore an important prerequisite to safely apply a refactoring of this kind.

5 Proof-of-Concept: Integration of Law Tests into Testing Frameworks and Rule Markup Languages

For our proof-of-concept implementation we use an extended ISO-Prolog-related scripting syntax (called Prova for PROlog + JaVA). In Prova, variables start with a uppercase letter, e.g. X, Y, Z, a constant/individual with a lowercase letter, e.g. a, b, c and a query is written as a function $: -solve(...)$ or $: -eval(...)$.

A *law test script* consists of a unique test ID denoted by `testcase(<ID>)`, optional input assertions such as input facts and test rules, an optional pre-test defining the test queries and variable bindings `testSuccess(<Test Name>,<Optional Message for Junit>)`, a post-test rule `testFailure(<Test Name>,<Message>)` and a `runTest` rule which is used by the meta program which implements the test axioms. The conditional law tests are interpreted by using the test rules to derive the success or failure of each test case.

Example 5.1. Rulebase with Test Case

```
a():-not(b()).
:-solve(test(./examples/pre-tc1.test)).

a():-not(b()).
c():-not(c()).
:-solve(test(./examples/post-tc1.test)).
```

Test Case: $tc1 := \{a()? => true\}$

```
testcase("tc1.test"). % id
% positive test with success message for JUnit report
testSuccess("test1","succeeded"):-
                testcase(tc1.test),a().
% negative test with failure message for Junit report
testFailure("test1","Relevance law violated"):-
                 not(testSuccess("test1",Message)).
% define the active tests - used by meta-program
runTest("./examples/pre-tc1.test"):-
                      testSuccess("test 1",Message).
runTest("./examples/post-tc1.test"):-
                      testFailure("test 1",Message).
```

The example shows the pre- and post test programs for testing the Relevance law. The test $test1$ first runs the pre-test deriving $a()$, and then tests with the post-test can no longer be derived, i.e. Relevance is violated. The second argument in the testSuccess/testFailure heads specifies a success resp. failure message which is used for reporting, e.g. to create a JUnit test report. The $runTest$ rule with the test case ID as argument is evaluated by the meta-program implementing the $test$ functionality. A test case might define several tests ($testSuccess/testFailure$).

Test cases can be bundled into test suites which are also represented as LPs consisting of a test suite ID denoted by $test\_suite(< name >)$. and a list of test cases referenced by their URI $test\_case(< URI >)$.. For automated testing we have implemented support for JUnit which runs the tests, creates a final test report which gives the test coverage and the failure and success messages of the executed tests.

A Test coverage measurement has been integrated into the test framework. (Paschke 2006; Paschke et al. 2006). The implemented coverage meta-program implements different functions to compute e.g. the minimalized substitution of two terms, the instances of clauses under the instance order, the lgg of two clauses, the subsumption of clauses, the generalized subsumption of two clause sets, the relative generalization and the coverage. The results are use to create an automated test coverage report.

To support platform-independent rule interchange we have integrated test case and updates into the Rule Markup Language (RuleML 1.02) [4]. It provides a rich syntax for derivation and reaction rules and supports different LP classes such as datalog, hornlog (with naf), extended (with neg) disjunctive, FOL.

A markup serialization syntax for test performatives/test suites/test cases and knowledge updates has been implemented as an extension to Reaction RuleML 1.02 with the following constructs:

```
Test ::= (vvi | Entails | TestSuite | TestCase)*
vvi ::= (Entails | TestSuite | TestCase)
TestSuite ::= (testbase | Assert | Consult)* ,
                  (vvi | Entails | TestSuite | TestCase)*
TestCase ::= (do | Assert | Retract | Update |
             Query | Answer | Test | Send | Receive |
             Action | Consult) , (expectedResult | Answer)
```

[4] `http://ruleml.org/1.02`

The vvi (*v*erification, *v*alidation, *i*ntegrity) tests of a $TestSuite$ are a set of entailments $Entails$, nested $TestSuite$s or single $TestCase$s. A $TestSuite$ consists of one or more test bases $testbase$ with assertions $Assert$ or consulted imports $Consult$ and one or more vvi tests. A single $TestCase$ consist of a doable performative, such as a $Query$ and the predefined $expectedResult$ such as an $Answer$. The $Test$ performative executes the tests it contains.

The optional $@style$ attribute can be used to assign meta-information about the intended testing protocol of a test performative, or the semantics (referenced semantic profile $Profile$) of a test suite, or a test case, e.g. to select a particular semantics from a target inference engine that implements several LP semantics.

```
<Test>
 <TestSuite @style="semantics:STABLE">
    <testbase> <Assert>
          ...test assertions
    </Assert> </testbase>
    <vvi> <TestCase>
      <do><Query>
           ... test query
      </Query> </do>
      <expectedResult> <Answer>
            ... expected answer
        </Answer> </expectedResult>
    </TestCase></vvi>
 </TestSuite>
<Test>
```

6 Related Work

V&V of KB systems and in particular rule-based systems such as LPs with Prolog interpreters have received much attention from the mid ’80s to the early ’90s, see e.g. (Antoniou et al. 1998). Criteria for verification and validation range from e.g. structural checks for relevance, redundancy and reachability to semantics tests for completeness and consistency. For a survey see (Preece 2001). Different approaches and methodologies to V&V of rule-based systems have been proposed in the literature such as model checking, code inspection, *operational debugging* (Byrd 1980) via instrumenting the rule base and exploring the execution trace, *tabular methods* (van Melle, Shortliffe, and Buchanan 1984), which pairwise compare the rules of the rule base to detect relationships among premises and conclusions, methods based on*formal graph theory* (Ramaswamy, Sarkar, and sho Chen 1997) or *Petri Nets* (He et al. 1999) which translate the rules into graphs or Petri nets, methods based on *declarative debugging* (Shapiro 1983) which build an abstract model of the LP and navigate through it or methods based on *algebraic interpretation* (Laita et al. 1999) which transform a KB into an algebraic structure, e.g. a boolean algebra which is then used to verify the KB.

Simple operational debugging approaches which instrument the rules and explore its execution trace place a huge cognitive load on the user, who needs to analyze each step of the conclusion process and needs to understand the structure of the rule system under test. On the other hand, typical heavy-weight V&V methodologies are often not suitable for rule-based systems, because they induce high costs of change and do not facilitate evolutionary modelling of rule bases. Model-checking techniques and methods based e.g. on algebraic-, graph- or Petri-net-based interpretations are computationally very costly, inapplicable for expressive rule languages and presuppose a deep understanding of both domains, i.e. of the testing language / models and of the rule language and the rule inferences.

Tests are particular suitable when rule bases grow larger and more complex and are maintained, possibly distributed and interchanged, by different users. Due to their inherent simplicity, tests better support different roles which are involved during the engineering process and give an expressive but nevertheless easy to use testing language since tests are written in the same language as the tested rule program. Research has also been directed at the test-driven refinement of rule bases (Dietrich and Paschke 2005), on the automatic generation of test cases, and test coverage measurement computations for test suites. (Denney 1991; Luo et al. 1992; Paschke 2006)

7 Conclusion and Future Work

In the context of rule interchange and selection of target inference engines we have proposed a testing methodology exploiting law tests to analyze the implementation specifics of an inference engine with respect to general properties of the semantics. The approach assesses the suitability of a target environment for the interchanged LP even if no further meta-information about the execution environment is available. This helps to safeguard rule interchange.

The concepts introduced in the paper have been implemented in an extended Prolog KR (called Prova) by integrating the law-test-driven development into the common testing framework JUnit with support for automated Ant task execution. This adds tool-support for law-test-driven development of logic programs.

Finally, we have introduced a concrete RuleML-based syntax for serialization of law tests enabling standardized platform-independent rule interchange.

We plan to extend our approach in several ways,

- automated law test instance generation from compact law statements,
- applying automated refactoring,
- larger suites of law test to test logic programs and inference engines,
- in order to provide them as open services on the web, a deeper and more fine-grained vocabulary to annotate rule engines and logic programs/law tests.

Our vision is to reach a similar market maturity of test-drive development in the agile development of rule programs and rule-based knowledge engineering, as is available in Extreme Programming for imperative software development.

8 Acknowledgments

This work has been partially supported by the Innoprofile Transfer project “Corporate Smart Content” funded by the German Federal Ministry of Education and Research (BMBF) and the BMBF Innovation Initiative for the New German Länder - Entrepreneurial Regions.

References

Antoniou, G.; van Harmelen, F.; Plant, R.; and Vanthienen, J. 1998. Verification and validation of knowledge-based systems: Report on two 1997 events. *AI Magazine* 19(3):123–126.

Apt, K. R., and Bol, R. 1994. Logic programming and negation: A survey. *JOURNAL OF LOGIC PROGRAMMING* 19:9–71.

Byrd, L. 1980. Understanding the control flow of prolog programs. In *Logic Programming Workshop*.

Denney, R. 1991. Test-case generation from prolog-based specifications. *IEEE Software* 8(2):49–57.

Dietrich, J., and Paschke, A. 2005. On the test-driven development and validation of business rules. In Kaschek, R.; Mayr, H. C.; and Liddle, S. W., eds., *Information Systems Technology and its Applications, 4th International Conference ISTA'2005, 23-25 May, 2005, Palmerston North, New Zealand*, volume 63 of *LNI*, 31–48. GI.

Dix, J. 1995a. Semantics of logic programs: Their intuitions and formal properties. *Logic, Action and Information, Essays on Logic in Philosophy and Artificial Intelligence (DeGruyter, 1995)* 241–327.

Dix, J. 1995b. A classification theory of semantics of normal logic programs: I. strong properties. *Fundam. Inform.* 22(3):227–255.

Dix, J. 1995c. A classification theory of semantics of normal logic programs: Ii. weak properties. *Fundam. Inform.* 22(3):257–288.

He, X.; Chu, W. C.; Yang, H.; and Yang, S. J. H. 1999. A new approach to verify rule-based systems using petri nets. In *COMPSAC*, 462–467. IEEE Computer Society.

ISO/IEC/IEEE 29119-1. 2013. ISO/IEC/IEEE 29119-1:2013 Software and systems engineering–Software testing–Part 1: Concepts and definitions. ISO 29119.

Kraus, S.; Lehmann, D. J.; and Magidor, M. 1990. Nonmonotonic reasoning, preferential models and cumulative logics. *Artif. Intell.* 44(1-2):167–207.

Laita, L. M.; Roanes-Lozano, E.; Maojo, V.; and Ledesma, L. d. 1999. Computer algebra based verification and knowledge extraction in rbs - application to medical fitness criteria. In *Collected Papers from the 5th European Symposium on Validation and Verification of Knowledge Based Systems - Theory, Tools and Practice*, EUROVAV '99, 53–65. Deventer, The Netherlands, The Netherlands: Kluwer, B.V.

Luo, G.; Bochmann, G.; Sarikaya, B.; and Boyer, M. 1992. Control-flow based testing of prolog programs. In *Software Reliability Engineering, 1992. Proceedings., Third International Symposium on*, 104–113.

Minker, J. 1993. An overview of nonmonotonic reasoning and logic programming. *Journal of Logic Programming, Special Issue* 17:95–126.

Paschke, A.; Dietrich, J.; Giurca, A.; Wagner, G.; and Lukichev, S. 2006. On self-validating rule bases. In *Proceedings of the 2nd International Workshop on Semantic Web Enabled Software Engineering (SWESE 2006).*

Paschke, A. 2006. Verification, validation, integrity of rule based policies and contracts in the semantic web. In *Proceedings of the 2nd International Semantic Web Policy Workshop (SWPW'06), Nov. 5-9, 2006, Athens, GA, USA*.

Paschke, A. 2014. Reaction ruleml 1.0 for rules, events and actions in semantic complex event processing. In *Rules on the Web. From Theory to Applications - 8th International Symposium, RuleML 2014, Co-located with the 21st European Conference on Artificial Intelligence, ECAI 2014, Prague, Czech Republic, August 18-20, 2014. Proceedings*, 1–21.

Preece, A. 2001. Evaluating verification and validation methods in knowledge engineering. *MICRO-LEVEL KNOWLEDGE MANAGEMENT* 123–145.

Ramaswamy, M.; Sarkar, S.; and sho Chen, Y. 1997. Using directed hypergraphs to verify rule-based expert systems. *Knowledge and Data Engineering, IEEE Transactions on* 9(2):221–237.

Shapiro, E. Y. 1983. *Algorithmic Program Debugging*. MIT Press.

Shepherdson, J. C. 1991. Unsolvable problems for sldnf resolution. *J. Log. Program.* 10(1):19–22.

van Melle, W.; Shortliffe, E. H.; and Buchanan, B. G. 1984. Emycin: A knowledge engineer's tool for constructing rule-based expert systems. In *Rule-based expert systems*. Addison-Wesley.

On Stochastic Belief Revision and Update and their Combination

Gavin Rens
Centre for Artificial Intelligence Research,
University of KwaZulu-Natal, School of Mathematics, Statistics and Computer Science and
CSIR Meraka, South Africa
Email: gavinrens@gmail.com

Abstract

I propose a framework for an agent to change its probabilistic beliefs when a new piece of propositional information α is observed. Traditionally, belief change occurs by either a revision process or by an update process, depending on whether the agent is *informed* with α in a static world or, respectively, whether α is a 'signal' from the environment due to an event occurring. Boutilier suggested a unified model of qualitative belief change, which "combines aspects of revision and update, providing a more realistic characterization of belief change." In this paper, I propose a unified model of *quantitative* belief change, where an agent's beliefs are represented as a probability distribution over possible worlds. As does Boutilier, I take a dynamical systems perspective. The proposed approach is evaluated against several rationality postulated, and some properties of the approach are worked out.

Information acquired can be due to evolution of the world or revelation about the world. That is, one may notice via some 'signal' generated by the changing environment that the environment has changed, or, one may be informed by an independent agent in a static environment that some 'fact' holds.

In the present work, I deal with belief change of agents who handle uncertainty by maintaining a probability distribution over possible situations. The agents in this framework also have models for nondeterministic events, and noisy observations. Noisy observation models can model imperfect sensory equipment for receiving environmental signals, but they can also model untrustworthy informants in a static world.

In this paper, I provide the work of Boutilier (1998) as background, because it has several connections with and was the seed for the present work. However, I do not intend simply to give a probabilistic version of his Generalized Update Semantics. Whereas Boutilier (1998) presents a model for unifying qualitative belief revision and update, I build on his work to present a unified model of belief revision and update in a stochastic (probabilistic) setting. I also take a dynamical systems perspective, like him. Due to my quantitative approach, an agent can maintain a probability distribution over the worlds it believes possible, using an *expectation* semantics of change. This is in contrast to Boutilier's "generalized update" approach, which takes a most-plausible event and most-plausible world approach. Finally, my proposal requires a trade-off factor to mix the changes in probability distribution over possible worlds brought about due to the probabilistic belief revision process and, respectively, the probabilistic belief update process. Boutilier's model has revision and update more tightly coupled. For this reason, his approach is better called "unified" while mine is called "hybrid".

The belief change community does not study probabilistic belief *update*; it is studied almost exclusively in frameworks employing Bayesian conditioning – for modeling events and actions in dynamical domains (e.g., DBNs, MDPs, POMDPs) (Koller and Friedman, 2009; Poole and Mackworth, 2010, e.g.). The part of my approach responsible for updating stays within the Bayesian framework, but combines the essential elements of belief update with *unobservable events* and belief update as partially observable Markov decision process (POMDP) *state estimation*.

On the other hand, there is plenty of literature on probabilistic belief *revision* (Voorbraak, 1999; Grove and Halpern, 1998; Kern-Isberner, 2008; Yue and Liu, 2008, e.g.). The subject is both deep and broad. There is no one accepted approach and to argue which is the best is not the focus of this paper. I shall choose one reasonable method for probabilistic belief revision suitable to the task at hand.

In the first section, Boutilier's 'generalized update' is reviewed. Then, in the next section, I introduce stochastic update and stochastic revision, culminating in the 'hybrid stochastic belief change' (HSBC) approach. The final section presents an example inspired by Boutilier's article (1998) and analyses the results.

Some proofs of propositions are omitted to save space; they are available on request.

Boutilier's Generalized Update

I use Boutilier's notation and descriptions, except that I am more comfortable with α and β to represent sentences, instead of A and B. It is assumed that an agent has a deductively closed belief set K, a set of sentences drawn from some logical language reflecting the agent's beliefs about the current state of the world. For ease of presentation, I assume a logically finite, classical propositional language, denoted L (L_{CPL} in Boutilier (1998)), and consequence operation Cn. The belief set K will often be generated by some finite knowledge base KB (i.e., $K = Cn(KB)$). The identically true and false propositions are denoted $\top$ and $\bot$, respectively. Given a set of possible worlds W (or valuations over L) and $\alpha \in L$, the set of α-worlds, that is, the elements of W satisfying α, is denoted by $||\alpha||$. The worlds satisfying all sentences in a set K is denoted $||K||$.

Update

Given a belief set K, an agent will often observe a change in the world α, requiring the agent to change K. This is the *update* of K by α, denoted $K^{\diamond}_{\alpha}$.

"$||KB||$ represents the set of possibilities we are prepared to accept as the actual state of affairs. Since observation α is the result of some change in the actual world, we ought to consider, for each possibility $w \in ||KB||$, the most plausible way (or ways) in which w might have changed in order to make α true. That is, we want to consider the most plausible evolution of world w into a world satisfying the observation α. To capture this intuition, Katsuno and Mendelzon (1991) propose a family of preorders $\{\leq_w \mid w \in W\}$, where each $\leq_w$ is a reflexive, transitive relation over W. We interpret each such relation as follows: if $u \leq_w v$ then u is at least as plausible a change relative to w as is v; that is, situation w would more readily evolve into u than it would into v.

Finally, a faithfulness condition is imposed: for every world w, the preorder $\leq_w$ has w as a minimum element; that is, $w <_w v$ for all $v \neq w$. Naturally, the most plausible candidate changes in w that result in α are those worlds v satisfying α that are minimal in the relation $\leq_w$. The set of such minimal α-worlds for each relation $\leq_w$, and each $w \in ||KB||$, intuitively capture the situations we ought to accept as possible when updating KB with α," (Boutilier, 1998, p. 9). In other words,

$$||KB \diamond \alpha|| = \bigcup_{w \in ||KB||} \{Min(\alpha, \leq_w)\},$$

where $Min(\alpha, \leq_w)$ specifies the minimal α-worlds with respect to the preorder $\leq_w$. Then $K^{\diamond}_{\alpha} = Cn(KB \diamond \alpha)$, where K is the belief set associated with KB.

Revision

Given a belief set K, an agent will often obtain information α in a static world, which must be incorporated into K. This is the *revision* of K by α, denoted K^{*}_{α}.

The AGM theory of belief revision (Alchourrón, Gärdenfors, and Makinson, 1985) provides a set of guidelines, in the form of the postulates, governing the process. "Unfortunately, while the postulates constrain possible revisions, they do not dictate the precise beliefs that should be retracted when α is observed. An alternative model of revision, based on the notion of epistemic entrenchment (Gärdenfors, 1988), has a more constructive nature," (Boutilier, 1998, p. 6).

"Semantically, an entrenchment relation (hence a revision function) can be modeled using an ordering on possible worlds reflecting their relative plausibility (Grove, 1988; Boutilier, 1994). However, rather than use a qualitative ranking relation, we adopt the presentation of (Spohn, 1988; Goldszmidt and Pearl, 1992) and rank all possible worlds using a κ-ranking. Such a ranking $\kappa : W \to N$ assigns to each world a natural number reflecting its plausibility or degree of believability. If $\kappa(w) < \kappa(v)$ then w is more plausible than v or more consistent with the agent's beliefs. We insist that $\kappa^{-1}(0) \neq \emptyset$, so that maximally plausible worlds are assigned rank 0. These maximally plausible worlds are exactly those consistent with the agent's beliefs; that is, the epistemically possible worlds according to K are those deemed most plausible in κ (see Spohn (1988) for further details). We sometimes assume κ is a partial function, and loosely write $\kappa(w) = \infty$ to mean $\kappa(w)$ is not defined (i.e., w is not in the domain of κ, or w is impossible)," (Boutilier, 1998, p. 6).

A κ-ranking captures the entrenchment of the agent's beliefs in its belief set K. This entrenchment determines how K will be revised when the agent receives new information / makes an observation α. κ induces a belief set as follows.

$$K = \{\alpha \in L \mid \kappa^{-1}(0) \subseteq ||\alpha||\}.$$

Due to the ranking or entrenchment of knowledge provided by κ, κ is considered an *epistemic state*.

"In other words, the set of most plausible worlds (those such that $\kappa(w) = 0$) determine the agent's beliefs. The ranking κ also induces a revision function: to revise by α an agent adopts the most plausible α-worlds as epistemically possible," (Boutilier, 1998, p. 6).

Let $W_i = \{w \in W \mid \kappa(w) = i\}$. And let $Min(\alpha, \kappa)$ be the set W_i with the least i such that for all $w^i \in W_i, w_i \models \alpha$. Then

$$K^{*}_{\alpha} := \{\beta \in L \mid Min(\alpha, \kappa) \subseteq ||\beta||\}.$$

In words, the belief set revised by α contains all those sentences entailed by the set of worlds with the same rank, where that rank is the least such that they are all α-worlds.

Generalized Update

As explained in the introduction, my intention with this paper is not to give a probabilistic version of the Generalized Update approach (Boutilier, 1998). For completeness, however, I sketch the approach here covering the approach in detail would take up unnecessary space without lending much insight into my Hybrid Stochastic Belief Change approach.

Boutilier motivates the need for a generalized update method which includes revision, by claiming that KM update (Katsuno and Mendelzon, 1991) is insufficient. He provides the following example adopted from Moore (1990). Suppose you want to test whether the contents of a beaker are chemically acid or base. If it is acid, a piece of litmus paper will turn red, if base, the paper will turn blue. Suppose that the test has not yet been performed, but you believe that the contents in the beaker are acidic. When the litmus paper is dipped into and pulled out of the beaker, the paper turns blue, indicating a basic compound. "Unfortunately, the KM theory does not allow this to take place. [...] One is forced to accept that, if the contents were acidic (in which case it should turn red), some extraordinary change occurred (the test failed, the contents of the beaker were switched, etc.). [...] Of course, the right thing to do is simply admit that the beaker did not, in fact, contain an acid—the agent should *revise* its beliefs about the contents of the beaker," (Boutilier, 1998, p. 13).

Boutilier adopts an event-based approach where a set of events E is assumed. These events are allowed to be nondeterministic, and each possible outcome of an event is ranked according to its plausibility via a κ-ranking. "As in the original event-based semantics, we will assume each world has an event ordering associated with it that describes the plausibility of various event occurrences at that world," (Boutilier, 1998, p. 14).

A *generalized update model* is then defined as $\langle W, \kappa, E, \mu\rangle$, where W is a set of worlds (the agent's epistemic state), κ is a ranking over W, E is a set of events (mappings over W), and μ is an event ordering (a set of mappings over E).

As with KM update, updates usually occur in response to some observation, with the assumption that something occurred to cause this observation. After observing α, an agent should adjust its beliefs by considering that only the most plausible transitions leading to α actually occurred. The set of possible α-transitions are those transitions leading to α-worlds. The most plausible α-transitions are those possible α-transitions with the minimal κ-ranking. Given that α has actually been observed, an agent should assume that one of these transitions describes the actual course of events. The worlds judged to be epistemically possible are those that result from the most plausible of these transitions.

Boutilier (1998) has a proposition that states that generalized belief update as described above is equivalent to "first determining the predicted updated ranking $\kappa^\diamond$ followed by standard (AGM) revision by α with respect to $\kappa^\diamond$," (Boutilier, 1998, p. 16). $\kappa^\diamond$ is determined by taking the worlds in the current possible worlds $||K||$ (induced from κ) and shifting them to all possible worlds given all possible transitions given all possible events (the actual event is unknown), taking into account the relevant plausibility rankings.

Stochastic Belief Change

I now consider agents who deal with uncertainty by maintaining a probability distribution over possible situations (worlds) they could be in. Let a *belief state* b be defined as the set $\{(w, p) \mid w \in W, p \in [0, 1]\}$, where $\sum_{(w,p)\in b} p = 1$. The probability of being in w is denoted $b(w)$. That is, b is a probability distribution over all the worlds in W. In the hybrid stochastic belief change (HSBC) framework, an agent maintains a belief state, which changes as new information is received or observed.

An agent is assumed to have a model of how the world works.

Definition 1. *The* stochastic belief change model M *has the form* $\langle W, \varepsilon, T, E, O, os\rangle$, *where*

- *W is a set of possible worlds,*
- *ε is a set of events,*
- $T : (W \times \varepsilon \times W) \to [0, 1]$ *is a* transition function *such that for every $e \in \varepsilon$ and $w \in W$, $\sum_{w'\in W} T(w, e, w') = 1$ ($T(w, e, w')$ models the probability of a transition to world w', given the occurrence of event e in world w),*
- *E is the event likelihood function ($E(e, w) = P(e \mid w)$, the probability of the occurrence of event e in w),*
- $O : (L \times W) \to [0, 1]$ *is an* observation function *such that for every world w, $\sum_{\alpha\in\Omega} O(\alpha, w) = 1$ ($O(\alpha, w)$ models the probability of observing α in w), where $\Omega \subset L$ is the set of possible observations, up to equivalence, and where if $\alpha \equiv \beta$, then $O(\alpha, w) = O(\beta, w)$, for all worlds w.*[1]
- *os : $(\Omega \times W) \to [0, 1]$ ($os(\alpha, w)$ is the agent's ontic strength for α perceived in w.)*

Definition 2. $b(\alpha) := \sum_{w\in W, w\models\alpha} b(w)$.

Let $b^\circ_\alpha := b \circ \alpha$ so that we can write $b^\circ_\alpha(w)$, where $\circ$ is any update or revision operator.

Often, in the exposition of this paper, a world will be referred to by its truth vector. For instance, if the vocabulary is $\{q, r\}$ and $w_3 \models \neg q \wedge r$, then w_3 may be referred to as 01.

For parsimony, let $b = \langle p_1, \ldots, p_n\rangle$ be the probabilities that belief state b assigns to $w_1, \ldots, w_n$ where $\langle w_1, w_2, w_3, w_4\rangle = \langle 11, 10, 01, 00\rangle$, and $\langle w_1, w_2, \ldots, w_8\rangle = \langle 111, 110, \ldots, 000\rangle$.

Update

Transitions associated with the observation of α from a world w in the current belief state b_{cur} to a world w' could be caused by different events. According to Boutilier (1998), update can be defined as

$$b^{event}_{new} := \Big\{(w', p') \mid w' \in W, p' = \sum_{w\in W}\sum_{e\in\varepsilon} T(w, e, w')E(e, w)b_{cur}(w)\Big\}.$$

Because the actual event is unobservable/hidden, p' is the *expected* probability of reaching w', given the event probabilities.

In partially observable Markov decision process (POMDP) theory (Aström, 1965; Monahan, 1982; Lovejoy, 1991), events are actions chosen by the agent (and thus observable) and observations are hidden. Then, given current belief state b_{cur}, selected action a and observation o, the *state estimation function* is defined by

$$b^{pomdp}_{new} := \left\{(w', p') \mid w' \in W, p' = \frac{O(o, a, w')\sum_{w\in W} T(w, a, w')b_{cur}(w)}{P(o \mid a, b_{cur})}\right\},$$

where Ω is a set of observation objects and $O : (\Omega \times A \times W) \to [0, 1]$ is an *observation function*, such that for every a and w', $\sum_{o\in\Omega} O(o, a, w') = 1$. $O(o, a, w')$ models the probability of perceiving o in arrival world w', given the execution of some action $a \in A$. Note that $P(o \mid a, b_{cur})$ is a normalizing constant.

[1] $\equiv$ denotes logical equivalence.

But what is the probabilistic update, given new information/evidence α? I suggest that α is the (overt) 'signal' generated by the (covert) event. An important question is, When is α received – in the current/departure world (w_c) or in the new/arrival world (w_n)? Although it is not clear to me, in POMDP theory, observations are always assumed to be received in the arrival world – I shall assume the same.

In the present framework, actions are not selected by the agent, but by nature. In other words, actions are considered to be events occurring in the environment, uncontrollable by the agent. Further, at the present stage of research, I shall assume that the agent has a less detailed observation model, that is, an agent only knows $O(\alpha, w_n)$, the probability of perceiving α in arrival world w_n (defined in Def. 1). Hence, I propose to weight $b^{event}_{new}(w')$ by $O(\alpha, w_n)$ when receiving new information α and one knows that one's belief state should be *updated* (due to an evolving world). Then we can define

Definition 3.

$$b \diamond \alpha := \{(w', p') \mid w' \in W, p' = \frac{1}{\gamma} O(\alpha, w') \sum_{w \in W} \sum_{e \in \varepsilon} T(w, e, w') E(e, w) b(w)\},$$

where γ is a normalizing factor.

As far as I know, no-one has proposed rationality postulates for probabilistic update. The reason is likely due to probabilistic update being defined in terms of standard probability theory. The axioms of probability theory have been argued to be rational for several decades (although it is not without its detractors).

The following basic postulates for my probabilistic belief update are proposed. (Unless stated otherwise, it is assumed that α is logically satisfiable, i.e., $\vdash \neg\alpha$ is false.)

($P^\diamond 1$) $b^\diamond_\alpha$ is a belief state iff not $\vdash \neg\alpha$

($P^\diamond 2$) $b^\diamond_\alpha(\alpha) = 1$

($P^\diamond 3$) If $\alpha \equiv \beta$, then $b^\diamond_\alpha = b^\diamond_\beta$

Proposition 1. *If $b^\diamond_\alpha(\alpha) > 0$, it is not necessary that $b^\diamond_\alpha(\alpha) = 1$.*

Proof. Let the vocabulary be $\{q, r\}$. Let $b = \langle 0.4, 0, 0.1, 0.5\rangle$. Let there be only one event e. Let the transition function be specified as $T(11, e, 11) = 0.5$, $T(11, e, 10) = 0.5$, $T(10, e, 01) = 1$, $T(01, e, 00) = 1$, $T(00, e, 11) = 1$. Let $E(e, w) = 1$ for all $w \in W$. Let the evidence be q. Let $O(q, 11) = 0.2$, $O(q, 10) = 0$, $O(q, 01) = 0$, $O(q, 00) = 0.3$. Then applying operation $\diamond$ to b produces $b^\diamond_q = \langle 0.82, 0, 0, 0.18\rangle$. Hence, $b^\diamond_q(q) = 0.82 \neq 1$. □

Although the following proposition is mostly negative, the reader will soon see that constraining the stochastic belief change model to be 'rational', the negative postulates become positive.

Proposition 2. *Postulate ($P^\diamond 3$) holds, while ($P^\diamond 1$) and ($P^\diamond 2$) do not hold.*

Definition 4. *We say event e is* event-rational *when for all $w \in W$: there exists a w' such that $T(w, e, w') > 0$ iff $E(e, w) > 0$.*

Definition 5. *We say α is an e-*signal *when for all $w' \in W$: there exists a w such that $T(w, e, w') > 0$ iff $O(\alpha, w') > 0$.*

Definition 6. *We say a model M is* observation-rational *iff for all α, whenever $\vdash \neg\alpha$, $O(\alpha, w) = 0$ for all $w \in W$.*

The proposition below says that if one is rational w.r.t. observations and w.r.t. some event, and α is a signal produced by that event, then updating on α is defined.

Proposition 3. *If M is observation-rational, there exists an event $e \in \varepsilon$ which is event-rational and α is an e-signal, then $b^\diamond_\alpha$ is a belief state iff not $\vdash \neg\alpha$ (i.e., then ($P^\diamond 1$) holds).*

($P^\diamond 2$) does not hold under the antecedents of Proposition 3. Another definition is required as qualification:

Definition 7. *We say evidence α is* trustworthy *iff for all $w \in W$, if $w \not\models \alpha$, then $O(\alpha, w) = 0$.*

The proposition below says that if α is trustworthy, one is rational w.r.t. some event, and α is a signal produced by that event, then one should accept α in the updated belief state.

Proposition 4. *If α is trustworthy, there exists an event $e \in \varepsilon$ which is event-rational and α is an e-signal, then $b^\diamond_\alpha(\alpha) = 1$ (i.e., then ($P^\diamond 2$) holds).*

Proof. Not $\vdash \neg\alpha$ is assumed by default. Recall that $b^\diamond_\alpha(\alpha) = \sum_{w \in W, w \models \alpha} b^\diamond_\alpha(w)$. Referring to the ($\Leftarrow$) part of the proof of Proposition 3, $b^\diamond_\alpha(\alpha)$ is a belief state and thus $\sum_{w \in W} b^\diamond_\alpha(w) = 1$. Hence, for $b^\diamond_\alpha(\alpha)$ to be less than 1, there must exist a $w' \in W$ s.t. $w' \not\models \alpha$ and $b^\diamond_\alpha(w') > 0$. But then $O(\alpha, w') > 0$. Therefore, for ($P^\diamond 2$) not to hold, an agent needs to believe that $O(\alpha, w') > 0$ for some world w' where $w' \not\models \alpha$. But then α cannot be trustworthy (i.e., then ($P^\diamond 2$) holds. □

Definition 8 (Gärdenfors, 1988)**.** *A probabilistic belief change operation $\circ$ is said to be* preservative *iff for all belief states P and for all propositions α and β, if $P(\alpha) > 0$ and $P(\beta) = 1$, then $P^\circ_\alpha(\beta) = 1$.*

Proposition 5. *Operation $\diamond$ is not preservative.*

Definition 9. *We say evidence α is β-*trustworthy *if for all $w \in W$, if $w \not\models \beta$, then $O(\alpha, w) = 0$.*

Proposition 6. *If $b^\diamond_\alpha(\beta)$ is a belief state, $b(\beta) = 1$ and α is β-trustworthy, then $b^\diamond_\alpha(\beta) = 1$.*

Proof. $\sum_{w \in W} b^\diamond_\alpha(w) = 1$. Hence, for $b^\diamond_\alpha(\beta)$ to be < 1, there must exist a $w^\times \in W$ s.t. $w^\times \not\models \beta$ and $b^\diamond_\alpha(w^\times) > 0$. And because $b(\beta) = 1$, $b(w^\times) = 0$. So some probability mass must have been shifted from some β-world to the non-β-world $w^\times$.

By definition, $b^\diamond_\alpha(w^\times) = \frac{1}{\gamma} O(\alpha, w^\times) \sum_{w \in W} \sum_{e \in \varepsilon} T(w, e, w^\times) E(e, w) b(w)$. So for $b^\diamond_\alpha(w^\times)$ to be > 0, $O(\alpha, w^\times)$ must be > 0.

However, because α is β-trustworthy, $O(\alpha, w^\times) = 0$. Hence, $O(\alpha, w^\times) \not> 0$ and $b^\diamond_\alpha(\beta) \not< 1$. □

Proposition 7. $b^\diamond_{\alpha\wedge\beta} \neq (b^\diamond_\alpha)^\diamond_\beta$.

Proof. For instance, consider the example used in the proof of Proposition 1. Let α be q and let β be $q \wedge r$. Note that $\alpha \wedge \beta$ is then logically equivalent to $q \wedge r$. Let $O(q \wedge r, 11) = O(q \wedge r, 10) = 0.5$ and $O(q \wedge r, 01) = O(q \wedge r, 00) = 0$.

We know that $b^{\diamond}_{q} = \langle 0.82, 0, 0, 0.18\rangle$. Then $(b^{\diamond}_{q})^{\diamond}_{q \wedge r} = \langle 1, 0, 0, 0\rangle$. On the other hand, $b^{\diamond}_{q \wedge r} = \langle 0.875, 0, 0, 0.125\rangle$. □

Revision

Using Bayes' Rule[2] , $P(w_n \mid \alpha)$ can be determined:

$$P(w \mid \alpha) := \frac{O(\alpha, w)b(w)}{\sum_{w' \in W} O(\alpha, w')b(w')}.$$

Note that if $O(\alpha, w) = 0$, then $P(w \mid \alpha) = 0$.

It is not yet universally agreed what revision means in a probabilistic setting. In classical belief change, it is understood that if the new information α is consistent with the agent's current beliefs KB, then revision is equivalent to belief expansion (denoted $+$), where expansion is the logical consequences of $KB \cup \{\alpha\}$. It is mostly agreed upon that Bayesian conditioning corresponds to classical belief expansion. This is evidenced by Bayesian conditioning (BC) being defined only when $b(\alpha) \neq 0$. In other words, one could define revision to be

$$b \,\mathsf{BC}\, \alpha := \{(w, p) \mid w \in W, p = P(w \mid \alpha)\},$$

as long as $P(\alpha) \neq 0$.[3]

To accommodate cases where $b(\alpha) \neq 0$, that is, where α contradicts the agent's current beliefs and its beliefs need to be revised in the stronger sense, we shall make use of *imaging*. Imaging was introduced by Lewis (1976) as a means of revising a probability function. It has also been discussed in the work of, for instance, Gärdenfors (1988); Dubois and Prade (1993); Chhogyal et al. (2014); Rens and Meyer (2015). The following version of imaging must not be regarded as a fundamental part of the larger belief change framework presented here; it should be regarded as a placeholder or suggestion for the 'revision-module' of the framework. Informally, Lewis's original solution for accommodating contradicting evidence α is to move the probability of each world to its closest, α-world. Lewis made the strong assumption that every world has a *unique* closest α-world. More general versions of imaging allow worlds to have *several*, equally proximate, closest worlds.

Gärdenfors (1988) calls one of his generalizations of Lewis's imaging *general imaging*. Our method is also a generalization. We thus refer to his as *Gärdenfors's general imaging* and to our method as *generalized imaging* to distinguish them. It should be noted that these imaging methods are general revision methods and can be used in place of Bayesian conditioning for expansion. "Thus imaging is a more general method of describing belief changes than conditionalization," (Gärdenfors, 1988, p. 112).

Let $Min(\alpha, w, d)$ be the set of α-worlds closest to w measured with d. Formally,

$$Min(\alpha, w, d) := \{w' \in ||\alpha|| \mid \forall w'' \in ||\alpha||, d(w', w) \leq d(w'', w)\},$$

where $d(\cdot)$ is some acceptable measure of distance between worlds (e.g., Hamming or Dalal distance). It must also obey the faithfulness condition that for every world w, $d(w, w) < d(v, w)$ for all $v \neq w$.

Example 1. *Let the vocabulary be* $\{q, r, s\}$. *Let* α *be* $(q \wedge r) \vee (q \wedge \neg r \wedge s)$. *Suppose* d *is Hamming distance. Then*

$$\begin{aligned}
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 111, d) = \{111\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 110, d) = \{110\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 101, d) = \{101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 100, d) = \{110, 101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 011, d) = \{111\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 010, d) = \{110\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 001, d) = \{101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 000, d) = \{110, 101\}
\end{aligned}$$

Then generalized imaging (denoted GI) is defined as

Definition 10.

$$\begin{aligned}
b \,\mathsf{GI}\, \alpha := \big\{&(w, p) \mid w \in W, p = 0 \textit{ if } w \notin ||\alpha||,\\
&\textit{else } p = \sum_{\substack{w' \in W\\ w \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)|\big\}.
\end{aligned}$$

Example 2. *Continuing on Example 1: Let* $b = \langle 0, 0.1, 0, 0.2, 0, 0.3, 0, 0.4\rangle$.

$(q \wedge r) \vee (q \wedge \neg r \wedge s)$ *is abbreviated as* α.

$b^{\mathsf{GI}}_{\alpha}(111) = \sum_{\substack{w' \in W\\ 111 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(111)/|Min(\alpha, 111, d)| + b(011)/|Min(\alpha, 011, d)| = 0/1 + 0/1 = 0$.

$b^{\mathsf{GI}}_{\alpha}(110) = \sum_{\substack{w' \in W\\ 110 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(110)/|Min(\alpha, 110, d)| + b(100)/|Min(\alpha, 100, d)| + b(010)/|Min(\alpha, 010, d)| + b(000)/|Min(\alpha, 000, d)| = 0.1/1 + 0.2/2 + 0.3/1 + 0.4/2 = 0.7$.

$b^{\mathsf{GI}}_{\alpha}(101) = \sum_{\substack{w' \in W\\ 101 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(101)/|Min(\alpha, 101, d)| + b(100)/|Min(\alpha, 100, d)| + b(001)/|Min(\alpha, 001, d)| + b(000)/|Min(\alpha, 000, d)| = 0/1 + 0.2/2 + 0/1 + 0.4/2 = 0.3$.

And $b^{\mathsf{GI}}_{\alpha}(100) = b^{\mathsf{GI}}_{\alpha}(011) = b^{\mathsf{GI}}_{\alpha}(010) = b^{\mathsf{GI}}_{\alpha}(001) = b^{\mathsf{GI}}_{\alpha}(000) = 0$.

Notice how the probability mass of non-α-worlds is shifted to their closest α-worlds. If a non-α-world $w^{\times}$ with probability p has n closest α-worlds (equally distant), then each of these closest α-worlds gets p/n mass from $w^{\times}$.

Recall that in the proposed framework, agents have access to an observation model (formalized via an observation function $O(\cdot, \cdot)$). Given enough computational power and time, it would be irrational for an agent to ignore its observation model when revising its beliefs. Another proposed

[2] Bayes' Rule states (in the notation of this paper) that $P(w \mid \alpha) = P(\alpha \mid w)P(w)/P(\alpha)$ or $P(w \mid \alpha) = P(\alpha \mid w)P(w)/\sum_{w' \in W} P(\alpha \mid w')P(w')$.

[3] Note that in my notation, $b(\alpha)$ is equivalent to $P(\alpha)$.

definition for a stochastic belief revision operation based on imaging (denoted OGI) is thus

Definition 11.

$$b\,\mathsf{OGI}\,\alpha := \Big\{(w,p) \mid w \in W, p = \frac{O(\alpha,w)b^{\mathsf{GI}}_{\alpha}(w)}{\sum_{w' \in W} O(\alpha,w')b^{\mathsf{GI}}_{\alpha}(w')}\Big\},$$

where the denominator is a normalizing factor.

$b\,\mathsf{OGI}\,\alpha$ is *not* defined as

$$\Big\{(w,p) \mid w \in W, p = 0 \text{ if } w \not\in ||\alpha||,$$

$$\text{else } p = \sum_{\substack{w' \in W \\ w \in Min(\alpha,w',d)}} O(\alpha,w')b(w')/|Min(\alpha,w',d)|\Big\},$$

because α is assumed perceived in the new world w, not the old world w'.

Note that if $P(\cdot \mid \alpha)$ were used instead of $O(\alpha,\cdot)$, then OGI would be undefined whenever $b(\alpha) = 0$. But this is exactly the problem we want to avoid by using imaging. Another justification to rather use $O(\alpha,w)$ is that its value is positively correlated with $P(w \mid \alpha)$: If $O(\alpha,w) = 0$, then $P(w \mid \alpha) = 0$. If $P(w \mid \alpha) = 1$, then $O(\alpha,w)$ is maximal in b in the following sense: for all $w' \in W$, if $w' \neq w$, then either $b(w') = 0$ or $O(\alpha,w') = 0$, whereas $b(w) > 0$ and $O(\alpha,w) > 0$.

Note that the denominator my be zero, making OGI undefined in that case. I shall deal with this issue a little bit later.

Example 3. *Recall from Example 2 that* $b^{\mathsf{GI}}_{\alpha} = \langle 0, 0.7, 0.3, 0, 0, 0, 0, 0\rangle$ *and* α *is* $(q \wedge r) \vee (q \wedge \neg r \wedge s)$. *Let* $O(\alpha,w) = 0.3$, *say, for all* $w \in W$. *Then* $b^{\mathsf{OGI}}_{\alpha} = b^{\mathsf{GI}}_{\alpha}$. *Obviously, if the observation model carries no information with respect to* α, *then it has no influence on the agent's revised beliefs.*

Now let $O(\alpha,w) = 0.3$ *if* $w \models r$, *else* $O(\alpha,w) = 0.2$. *Then* $b^{\mathsf{OGI}}_{\alpha} = \langle 0.3 \times 0/0.23, 0.2 \times 0.7/0.23, 0.3 \times 0.3/0.23, 0.2 \times 0/0.23, 0.3 \times 0/0.23, 0.2 \times 0/0.23, 0.3 \times 0/0.23, 0.2 \times 0/0.23\rangle = \langle 0, 0.61, 0.39, 0, 0, 0, 0, 0\rangle$. *If the agent has an observation model telling it that* α *is more likely to be perceived in* r*-worlds than in* $\neg r$*-worlds, then when it receives* α, *the agent should be biased to believing that it is actually in an* r*-world. However, the agent was certain that it was in a* $\neg r$*-world when its belief state was* b. GI *thus pushes the agent to favour the* α*-worlds being* $\neg r$*-worlds. Hence, in this example there is tension between being in a* $\neg r$*-world (due to previous beliefs) and being in an* r*-world (due to the observation model).*

Definition 12.

$$b\,\mathsf{BCI}\,\alpha := \begin{cases} b\,\mathsf{BC}\,\alpha & \textit{if } b(\alpha) > 0 \\ b\,\mathsf{OGI}\,\alpha & \textit{if } b(\alpha) = 0 \end{cases}$$

I denote the *expansion* of belief state b on α as b^{+}_{α} (resp., probability function P on α as P^{+}_{α}) and delay its definition till later. $P_{\perp}$ is conventionally defined to be the absurd probability function which is defined to be $P_{\perp}(\delta) = 1$ for all $\delta \in L$.

Gärdenfors (1988) proposed six rationality postulates for probabilistic belief revision. (Unless stated otherwise, it is assumed that α is logically satisfiable, i.e., $\vdash \neg\alpha$ is false.)

1. P^{*}_{α} is a probability function
2. $P^{*}_{\alpha}(\alpha) = 1$
3. If $\alpha \equiv \beta$, then $P^{*}_{\alpha} = P^{*}_{\beta}$
4. $P^{*}_{\alpha} \neq P_{\perp}$ iff not $\vdash \neg\alpha$
5. If $P(\alpha) > 0$, then $P^{*}_{\alpha} = P^{+}_{\alpha}$
6. If $P^{*}_{\alpha}(\beta) > 0$, then $P^{*}_{\alpha\wedge\beta} = (P^{*}_{\alpha})^{+}_{\beta}$.

Instead of saying that the result of an operation is $P_{\perp}$, I simply say that the result is undefined. And by noting that the result of an operation is not a belief state if it is undefined, one can merge postulates 1 and 4. The stochastic belief revision postulates in my notation are thus

($P^{*}1$) b^{*}_{α} is a belief state iff not $\vdash \neg\alpha$
($P^{*}2$) $b^{*}_{\alpha}(\alpha) = 1$
($P^{*}3$) If $\alpha \equiv \beta$, then $b^{*}_{\alpha} = b^{*}_{\beta}$
($P^{*}4$) If $b(\alpha) > 0$, then $b^{*}_{\alpha} = b^{+}_{\alpha}$
($P^{*}5$) If $b^{*}_{\alpha}(\beta) > 0$, then $b^{*}_{\alpha\wedge\beta} = (b^{*}_{\alpha})^{+}_{\beta}$.

I now test OGI and BCI against each of the five postulates.

Recall that if the denominator in the definition of OGI is zero, it is undefined. To guarantee that OGI is defined, $\sum_{w' \in W} O(\alpha,w')b^{\mathsf{GI}}_{\alpha}(w')$ must be non-zero, that is, there must be at least one $w' \in W$ for which $O(\alpha,w')b^{\mathsf{GI}}_{\alpha}(w') > 0$. We know that when $w' \not\in ||\alpha||$, $O(\alpha,w')b^{\mathsf{GI}}_{\alpha}(w') = b^{\mathsf{GI}}_{\alpha}(w') = 0$.

Definition 13. *We say* α *is* weakly observable *iff there exists a* $w \in W$ *such that* $w \models \alpha$ *and* $O(\alpha,w) > 0$. *We say* α *is* strongly observable *iff for all* $w \in W$ *for which* $w \models \alpha$, $O(\alpha,w) > 0$.

Proposition 8. *When* $*$ *is* OGI, *postulate* ($P^{*}1$), *in general, does not hold, but does hold if evidence* α *is strongly observable.*

Proof. Firstly, observe that $b(w') = \sum_{|Min(\alpha,w',d)|} b(w')/|Min(\alpha,w',d)|$. Therefore,

$$\begin{aligned} 1 &= \sum_{w' \in w} b(w') \\ &= \sum_{w' \in w} \sum_{|Min(\alpha,w',d)|} b(w')/|Min(\alpha,w',d)| \\ &= \sum_{w' \in w, |Min(\alpha,w',d)|} b(w')/|Min(\alpha,w',d)| \\ &= \sum_{w' \in w, w \in W, w \in Min(\alpha,w',d)} b(w')/|Min(\alpha,w',d)| \\ &= \sum_{w \in W} \sum_{w' \in w, w \in Min(\alpha,w',d)} b(w')/|Min(\alpha,w',d)| \\ &= \sum_{w \in W} b^{\mathsf{GI}}_{\alpha}(w). \end{aligned}$$

Let $b = \langle 0, 0.1, 0, 0.2, 0, 0.3, 0, 0.4\rangle$ and α be $(q \wedge r) \vee (q \wedge \neg r \wedge s)$. Let $O(\alpha, 111) = 0.9$ and $O(\alpha,w) = 0$ for all $w \in W$, $w \neq 111$. (Notice that α is weakly observable.) From Example 2, we know that $b^{\mathsf{GI}}_{\alpha}(111) = 0$, implying that $b^{\mathsf{OGI}}_{\alpha}(111) = 0$, and one can deduce that $b^{\mathsf{OGI}}_{\alpha}(w) = 0$ for all $w \in W$, due to the specification of the observation model.

Now, let α be strongly observable: let $O(\alpha, 111) = O(\alpha, 110) = O(\alpha, 101) = 0.1$, else $O(\alpha, \cdot) = 0$. Then $b^{\mathsf{OGI}}_\alpha = \langle 0, 0.7, 0.3, 0, 0, 0, 0, 0\rangle$. In general, let $O(\alpha, w) > 0$ for all $w \in W$ for which $w \models \alpha$. By definition of GI, the probability mass of all non-α-worlds is shifted to their closest α-worlds; the total mass (of the α-worlds) thus remains 1. Hence, $b^{\mathsf{GI}}_\alpha(\alpha) = 1$ and there exists a $w' \models \alpha$ s.t. $b^{\mathsf{GI}}_\alpha(w') > 0$. Now, by definition of strong observability, $O(\alpha, w') > 0$. Therefore, $O(\alpha, w')b^{\mathsf{GI}}_\alpha(w') > 0$. And due to the normalizing effect of the denominator in the definition of OGI, b^{OGI}_α is a belief state. □

Proposition 9. *When $*$ is* OGI*, postulate (P^*2), in general, does not hold and does hold when α is strongly observable.*

Proof. This result follows directly from an understanding of the proof of Proposition 8. □

Proposition 10. *When $*$ is* OGI*, postulate (P^*3) holds.*

Proposition 11. *Let $*$ be* OGI*. If $+$ is* OGI*, postulate (P^*4) holds, otherwise it does not.*

Assuming (P^*4) holds, I consider whether (P^*5) holds only for two combinations of instantiations of $*$ and $+$.

Proposition 12. *When $*$ is* OGI *and $+$ is* OGI*, postulate (P^*5) does not hold.*

Proof. An instance is provided where $b^{\mathsf{OGI}}_\alpha(\beta) > 0$ and $b^{\mathsf{OGI}}_{\alpha\land\beta} \neq (b^{\mathsf{OGI}}_\alpha)^{\mathsf{OGI}}_\beta$.

Continuing with Example 3, where $b = \langle 0, 0.1, 0, 0.2, 0, 0.3, 0, 0.4\rangle$, α is $(q \land r) \lor (q \land \neg r \land s)$ and $b^{\mathsf{OGI}}_\alpha = b^{\mathsf{GI}}_\alpha = \langle 0, 0.7, 0.3, 0, 0, 0, 0, 0\rangle$. Let β be $q \land r$, then $b^{\mathsf{OGI}}_\alpha(\beta) = 0.7 > 0$. But $b^{\mathsf{OGI}}_{\alpha\land\beta} = b^{\mathsf{OGI}}_\beta = \langle 0, 1, 0, 0, 0, 0, 0, 0\rangle$ and $(b^{\mathsf{OGI}}_\alpha)^{\mathsf{OGI}}_\beta = \langle 0.3, 0.7, 0, 0, 0, 0, 0, 0\rangle$. □

Proposition 13. *When $*$ is* BCI*, postulate (P^*1) holds.*

Proof. It is known that Bayesian conditioning results in a belief state when the conditional is non-contradictory. □

Proposition 14. *When $*$ is* BCI*, postulate (P^*2) holds.*

Proof. By definition of BC, all non-α-worlds get zero probability and the probabilities of the remaining α-worlds are magnified to sum to 1. □

Proposition 15. *When $*$ is* BCI*, postulate (P^*3) holds.*

Proposition 16. *Let $*$ be* BCI*. If $+$ is* BC *or* BCI*, postulate (P^*4) holds, otherwise it does not.*

For the proof of the next proposition, a lemma is required.

Lemma 1. *Let $b(\alpha) > 0$. If $b^{\mathsf{BC}}_\alpha(\beta) > 0$, then $b(\alpha \land \beta) > 0$.*

Proof. Assume $b^{\mathsf{BC}}_\alpha(\beta) > 0$. Then there exists a $w^\beta \in W$ s.t. $w^\beta \models \beta$ and $b^{\mathsf{BC}}_\alpha(w^\beta) > 0$. By definition, $b^{\mathsf{BC}}_\alpha(w) = \frac{b(\alpha,w)}{b(\alpha)}$, implying $\frac{b(\alpha,w^\beta)}{b(\alpha)} > 0$. Hence, $b(w^\beta) > 0$ and $w^\beta \models \alpha$. But if $w^\beta \models \alpha$, then $w^\beta \models \alpha \land \beta$, and due to $b(w^\beta) > 0$, $b(\alpha \land \beta) > 0$. □

Proposition 17. *When $*$ is* BCI *and $+$ is* BC*, postulate (P^*5), in general, does not hold, but does hold when $b(\alpha) > 0$.*

Recall that a probabilistic belief change operation $\circ$ is *preservative* iff for all belief states b and for all propositions α and β, if $b(\alpha) > 0$ and $b(\beta) = 1$, then $b^\circ_\alpha(\beta) = 1$.

Proposition 18. *Operation* OGI *is not preservative, while* BCI *is preservative.*

Proof. OGI: Let the vocabulary be $\{q, r\}$ and $b = \langle 0, 0.5, 0.5, 0\rangle$. Let α be q and β be $q \leftrightarrow \neg r$. Then $b(\alpha) > 0$, $b(\beta) = 1$ and $b^{\mathsf{GI}}_\alpha = \langle 0.5, 0.5, 0, 0\rangle$. Let $O(q, w) = 1$ for all $w \in W$. Then $b^{\mathsf{OGI}}_\alpha = \langle 0.5, 0.5, 0, 0\rangle$ and $b^{\mathsf{OGI}}_\alpha(\beta) = 0.5$.[4]

BCI: $b^{\mathsf{BCI}}_\alpha(\beta) = b^{\mathsf{BC}}_\alpha(\beta)$. By assuming that $b(\alpha) > 0$ and $b(\beta) = 1$, one is implicitly assuming that if $w \models \alpha$ s.t. $b(\alpha) > 0$, then $w \models \beta$. This in turn implies that whenever $b^{\mathsf{BC}}_\alpha(w) > 0$, that $w \models \beta$. The latter is due to conditionalization: $\{w \in W \mid b^{\mathsf{BC}}_\alpha(w) > 0\}$ is a subset of $\{w \in W \mid w \models \alpha, b(w) > 0\}$. By ($P^*2$), $b^{\mathsf{BC}}_\alpha(\alpha) = 1$. But due to the fact that for all $w \in W$, if $b^{\mathsf{BC}}_\alpha(w) > 0$, then $w \models \beta$, it must then be the case that $b^{\mathsf{BC}}_\alpha(\beta) = 1$. □

Ontic and Epistemic Strength

Suppose there is a range of degrees for information being ontic (the effect of a physical action or occurrence) or epistemic (purely informative). I shall assume that the higher the information's degree of being ontic, the lower the epistemic status of that information. An agent has a certain sense of the degree to which a piece of received information is due to a physical action or event in the world. This sense may come about due to a combination of sensor readings and reasoning. If the agent performs an action and a change in the local environment matches the expected effect of the action, it can be quite certain that the effect is ontic information. If the agent receives the information from another agent (e.g., radio, through reading, a person speaking directly to the agent), then it should be clear to the agent that the information is epistemic and thus has a low degree of being ontic. If the agent's sensors show activity, but the agent knows that it did not presently perform an action with an effect matching its sensor readings, and if the readings do not reveal an epistemic source for the information, then the agent will have to infer from the present world conditions and the information received, or access learnt knowledge matching the present world conditions and the information received, the degree to which the information should be regarded as ontic. For instance, a person might stop talking just after you ask him/her to be quiet. Under particular conditions the person may stop talking due to your request and in other conditions he/she may have stopped talking anyway. Depending on the present world conditions, you might assign a higher (but not definitely certainty) or lower (but not definitely zero) degree of likelihood that the information (i.e., that the person stopped talking) is ontic. Or suppose you have been wearing dark glasses for one hour. You put them on due to the sky being clear and (too) bright. When you take your glasses off, it is not as bright as you thought it would be. So, has the ambient

[4] Here, d is Hamming distance.

brightness decreased due to changes in the weather, or does it only seem darker when you remove your glasses, due to some unknown physiological process? In this case, it would be convenient to consider the brightness/darkness information as being equally likely ontic and epistemic.

Recall from Definition 1 that $os(\alpha, w)$ indicates an agent's sense for the ontic strength of α received in w. We say that $os(\alpha, w) = 1$ when α is certainly ontic in w. When α is certainly epistemic in w, then $os(\alpha, w) = 0$. In fact, let the epistemic strength of α in w be defined as $es(\alpha, w) := 1 - os(\alpha, w)$.

Combining Update and Revision

I propose a way of trading off the probabilistic update and probabilistic revision defined earlier, using the notion of ontic strength.

The hybrid stochastic change of belief state b due to new information α with ontic strength (denoted $b \lhd \alpha$) is defined as

Definition 14.

$$b \lhd \alpha := \Big\{(w,p) \mid w \in W, p = \frac{1}{\gamma}\big[(1 - os(\alpha,w))b^*_\alpha(w) + os(\alpha,w)b^\diamond_\alpha(w)\big]\Big\},$$

where γ is a normalizing factor so that $\sum_{w\in W} b^\lhd_\alpha(w) = 1$.

Due to our assumption that α is observed in the *arrival* world, not the *departure* world, $os(\cdot)$ is applied to the *arrival* world.

Considering the rationality postulates presented so far for belief update and revision, one can naturally suggest the following postulates for their combination.

$(P^\lhd 1)$ $b^\lhd_\alpha$ is a belief state iff not $\vdash \neg\alpha$

$(P^\lhd 2)$ $b^\lhd_\alpha(\alpha) = 1$

$(P^\lhd 3)$ If $\alpha \equiv \beta$, then $b^\lhd_\alpha = b^\lhd_\beta$

Proposition 19. *Postulate $(P^\lhd 1)$ does not hold.*

Proposition 20. *Postulate $(P^\lhd 2)$ does not hold.*

Proof. $(P^\lhd 2)$ does not hold because $(P^\diamond 2)$ does not hold. □

Proposition 21. *Postulate $(P^\lhd 3)$ holds.*

Proof. $(P^\lhd 3)$ is holds because $(P^\diamond 3)$ and $(P^* 3)$ hold. □

Theorem 1. *If: the agent model M is observation-rational, α is trustworthy and strongly observable, there exists an event $e \in \varepsilon$ which is event-rational and α is an e-signal, then (i) $b^\lhd_\alpha$ is a belief state iff not $\vdash \neg\alpha$ (i.e., then $(P^\lhd 1)$ is true) and (ii) $b^\lhd_\alpha(\alpha) = 1$ (i.e., then $(P^\lhd 2)$ is true).*

Proof. Note that by Propositions 3 and 4, $(P^\diamond 1)$ and $(P^\diamond 2)$ hold. And recall that $(P^* 1)$ and $(P^* 2)$ are true when α is strongly observable (see Props. 8, 9, 13 and 14).

(i)$(P^\lhd 1)$ Given the antecedents of this proposition, we know by Proposition 4 that $b^\diamond_\alpha$ is defined iff not $\vdash \neg\alpha$. And by $(P^* 4)$, b^*_α is defined iff not $\vdash \neg\alpha$.

($\Rightarrow$) Assume $b^\lhd_\alpha$ is defined. So there exists a $w \in W$ s.t. $b^\lhd_\alpha(w) > 0$, that is, $\frac{1}{\gamma}\big[(1 - os(\alpha,w))b^*_\alpha(w) + os(\alpha,w)b^\diamond_\alpha(w)\big] > 0$. Thus, either $b^*_\alpha(w) > 0$ (while $1 - os(\alpha,w) > 0$) or $b^\diamond_\alpha(w) > 0$ (while $os(\alpha,w) > 0$) (or both), which implies that b^*_α resp. $b^\diamond_\alpha$ is defined. Therefore, not $\vdash \neg\alpha$.

($\Leftarrow$) Assume not $\vdash \neg\alpha$. Then b^*_α and $b^\diamond_\alpha$ are defined. This implied that there exists a $w \in W$ s.t. either $b^*_\alpha(w) > 0$ or $b^\diamond_\alpha(w) > 0$ (or both). Hence, $b^\lhd_\alpha(w) > 0$ and due to normalization in the definition of $\lhd$, $b^\lhd_\alpha$ is defined.

(ii)$(P^\lhd 2)$ $b^\lhd_\alpha(\alpha) = \sum_{w\in W, w\models\alpha} b^\lhd_\alpha(w) = \sum_{w\in W, w\models\alpha} \frac{1}{\gamma}\big[(1 - os(\alpha,w))b^*_\alpha(w) + os(\alpha,w)b^\diamond_\alpha(w)\big]$, where $\gamma = \sum_{w\in W}\big[(1 - os(\alpha,w))b^*_\alpha(w) + os(\alpha,w)b^\diamond_\alpha(w)\big]$. But by $(P^* 2)$ and $(P^\diamond 2)$, if $w \not\models \alpha$, then $b^*_\alpha(w) = 0$ and $b^\diamond_\alpha(w) = 0$. Hence, $\gamma = \sum_{w\in W, w\models\alpha}\big[(1 - os(\alpha,w))b^*_\alpha(w) + os(\alpha,w)b^\diamond_\alpha(w)\big]$. Therefore, $b^\lhd_\alpha(\alpha) = \sum_{w\in W, w\models\alpha} \frac{(1-os(\alpha,w))b^*_\alpha(w)+os(\alpha,w)b^\diamond_\alpha(w)}{\sum_{w\in W, w\models\alpha}\left[(1-os(\alpha,w))b^*_\alpha(w)+os(\alpha,w)b^\diamond_\alpha(w)\right]} = 1$. □

Although one cannot expect $\lhd$ to be preservative, due to probabilistic update not being preservative (Prop. 5), one can expect $\lhd$ to have preservative-like behaviour under particular conditions: Recall that α is defined to be *β-trustworthy* if for all $w \in W$, if $w \not\models \beta$, then $O(\alpha, w) = 0$.

Proposition 22. *If $b^\diamond_\alpha(\beta)$ is a belief state, $b^\lhd_\alpha(\beta)$ is a belief state, $b(\beta) = 1$, α is β-trustworthy and $*$ is* BCI*, then $b^\lhd_\alpha(\beta) = 1$.*

Proof. By Proposition 6, $b^\diamond_\alpha(\beta) = 1$, when α is β-trustworthy. By Proposition 18, $*$ is preservative when defined as BCI. Then, for all $w \in W$, if $w \not\models \beta$, then $b^\diamond_\alpha(\beta) = b^*_\alpha(\beta) = 0$. Hence, for all $w \in W$, if $w \not\models \beta$, $b^\lhd_\alpha(w) = 0$. Therefore, because $b^\lhd_\alpha(\beta)$ is a belief state,

$$\begin{aligned} b^\lhd_\alpha(\beta) &= 1 - b^\lhd_\alpha(\neg\beta) \\ &= 1 - \sum_{w\in w, w\models\neg\beta} b^\lhd_\alpha(w) \\ &= 1 - \sum_{w\in w, w\not\models\beta} b^\lhd_\alpha(w) \\ &= 1 - 0 \\ &= 1. \end{aligned}$$

□

Examples and Analysis

HSBC is now analyzed via examples. The example domain is adapted from one of the domains in the article of Boutilier (1998) – here though, worlds are associated with probabilities, not plausibility ranks. There are eight possible worlds, depending on whether a book B is inside the house (if it is not in the house, then it is assumed to be on the patio, adjacent to the lawn), whether the book is dry and whether the lawn-grass G is dry. There are three events: `rain` – it rains, `sprnk` – the sprinkler is on, and `null` – neither of these, the null event.[5] In Boutilier's example, events are deterministic; however, events in this paper are modeled to be stochastic, to better illustrate the behaviour of the framework.

To simplify calculations and to aid the reader in understanding the results, in the following examples, the agent will associate equal epistemic/ontic strength to a particular piece of information for all worlds (per example case). I shall compute the agent's new belief state for each of $os(\alpha, w) \in \{0, 0.25, 0.5, 0.75, 1\}$ (for all $w \in W$), for the two cases where α is $\neg\texttt{Dry}(G)$ and where α is $\neg\texttt{Dry}(G) \land \texttt{Dry}(B)$.

Boutilier models the agent's current (initial) epistemic state with the most plausible situation (rank 0) being $(\neg\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G))$ and the next plausible situation (rank 1) being $(\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G))$. I translate this as the agent having a belief state where $b(\neg\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G)) = 0.67$ and $b(\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G)) = 0.33$. Observe that in these examples, revision as OGI is equivalent to revision as BCI, because $b(\neg\texttt{Dry}(G)) = b(\neg\texttt{Dry}(G) \land \texttt{Dry}(B)) = 0$.

The HSBC model $M = \langle W, \varepsilon, T, E, O, os\rangle$ is now specified.

Let $w_1, \ldots, w_8$ refer to worlds

w_1:	$(\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G))$
w_2:	$(\texttt{Inside}(B), \texttt{Dry}(B), \neg\texttt{Dry}(G))$
w_3:	$(\texttt{Inside}(B), \neg\texttt{Dry}(B), \texttt{Dry}(G))$
w_4:	$(\texttt{Inside}(B), \neg\texttt{Dry}(B), \neg\texttt{Dry}(G))$
w_5:	$(\neg\texttt{Inside}(B), \texttt{Dry}(B), \texttt{Dry}(G))$
w_6:	$(\neg\texttt{Inside}(B), \texttt{Dry}(B), \neg\texttt{Dry}(G))$
w_7:	$(\neg\texttt{Inside}(B), \neg\texttt{Dry}(B), \texttt{Dry}(G))$
w_8:	$(\neg\texttt{Inside}(B), \neg\texttt{Dry}(B), \neg\texttt{Dry}(G))$

The events are $\varepsilon = \{\texttt{rain}, \texttt{sprnk}, \texttt{null}\}$.

The following probabilities are debatable; they should not be taken too seriously but serve to illustrate the framework.

$T(w_1, \texttt{null}, w_1) = 0.75$	$T(w_5, \texttt{null}, w_5) = 1$
$T(w_1, \texttt{null}, w_2) = 0.1$	$T(w_5, \texttt{null}, w_6) = 0$
$T(w_1, \texttt{null}, w_3) = 0.1$	$T(w_5, \texttt{null}, w_7) = 0$
$T(w_1, \texttt{null}, w_4) = 0.05$	$T(w_5, \texttt{null}, w_8) = 0$
$T(w_1, \texttt{rain}, w_1) = 0$	$T(w_5, \texttt{rain}, w_5) = 0$
$T(w_1, \texttt{rain}, w_2) = 0.75$	$T(w_5, \texttt{rain}, w_6) = 0.05$
$T(w_1, \texttt{rain}, w_3) = 0$	$T(w_5, \texttt{rain}, w_7) = 0.05$
$T(w_1, \texttt{rain}, w_4) = 0.25$	$T(w_5, \texttt{rain}, w_8) = 0.9$
$T(w_1, \texttt{sprnk}, w_1) = 0$	$T(w_5, \texttt{sprnk}, w_5) = 0$
$T(w_1, \texttt{sprnk}, w_2) = 0.8$	$T(w_5, \texttt{sprnk}, w_6) = 0.8$
$T(w_1, \texttt{sprnk}, w_3) = 0$	$T(w_5, \texttt{sprnk}, w_7) = 0.05$
$T(w_1, \texttt{sprnk}, w_4) = 0.2$	$T(w_5, \texttt{sprnk}, w_8) = 0.15$

$E(\texttt{null}, w_1) = 0.06$	$E(\texttt{null}, w_5) = 0.15$
$E(\texttt{rain}, w_1) = 0.31$	$E(\texttt{rain}, w_5) = 0.7$
$E(\texttt{sprnk}, w_1) = 0.63$	$E(\texttt{sprnk}, w_5) = 0.15$

$O(\neg\texttt{Dry}(G), w_1) = 0.05$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_1) = 0.5$
$O(\neg\texttt{Dry}(G), w_2) = 0.95$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_2) = 0.8$
$O(\neg\texttt{Dry}(G), w_3) = 0.05$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_3) = 0.1$
$O(\neg\texttt{Dry}(G), w_4) = 0.95$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_4) = 0.05$
$O(\neg\texttt{Dry}(G), w_5) = 0.05$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_5) = 0.6$
$O(\neg\texttt{Dry}(G), w_6) = 0.95$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_6) = 0.98$
$O(\neg\texttt{Dry}(G), w_7) = 0.05$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_7) = 0.2$
$O(\neg\texttt{Dry}(G), w_8) = 0.95$	$O(\neg\texttt{Dry}(G) \land \texttt{Dry}(B), w_8) = 0.15$

Recall that the current belief state is $b = \langle 0.33, 0, 0, 0, 0.67, 0, 0, 0\rangle$. The following is a list of resulting belief states $b' = b \lhd \neg\texttt{Dry}(G)$ for the specified ontic strengths.

$os(\cdot)$	$b \lhd \neg\texttt{Dry}(G)$
0.00	$\langle 0.00, 0.33, 0.00, 0.00, 0.00, 0.67, 0.00, 0.00\rangle$
0.25	$\langle 0.00, 0.32, 0.00, 0.02, 0.00, 0.53, 0.00, 0.13\rangle$
0.50	$\langle 0.00, 0.31, 0.00, 0.04, 0.00, 0.40, 0.00, 0.25\rangle$
0.75	$\langle 0.00, 0.30, 0.00, 0.06, 0.00, 0.26, 0.00, 0.38\rangle$
1.00	$\langle 0.00, 0.28, 0.00, 0.08, 0.01, 0.12, 0.00, 0.51\rangle$

Several behaviours can be noted: When the observation is completely epistemic, the probabilities of the two believed worlds are each shifted to their closest $\neg\texttt{Dry}(G)$-worlds. The more the agent considers the information to be ontic, the more its beliefs are spread out due to the nondeterminism of the events. Whether the observation is considered ontic or epistemic, the agent has a relatively strong belief (between 28% and 33%) that the book is inside and dry. However, in cases where the book is outside, there is a considerable shift in probability from the book being dry (w_6) to it being wet (w_8), as the agent moves towards an ontic mindset. One could perhaps argue that in an ontic mindset, the agent has access to event/transition information so as to reason about the causes of the book getting wet: it believes that there is a moderate to high likelihood that the book will get wet if it is on the patio, due to the sprinkler coming on or it starting to rain (explaining the wet-grass evidence).

The following is a list of resulting belief states $b' = b \lhd \neg\texttt{Dry}(G) \land \texttt{Dry}(B)$ for the specified epistemic strengths.

$os(\cdot)$	$b \lhd \neg\texttt{Dry}(G) \land \texttt{Dry}(B)$
0.00	$\langle 0.00, 0.29, 0.00, 0.00, 0.00, 0.71, 0.00, 0.00\rangle$
0.25	$\langle 0.00, 0.33, 0.00, 0.00, 0.03, 0.59, 0.00, 0.04\rangle$
0.50	$\langle 0.01, 0.37, 0.00, 0.00, 0.07, 0.47, 0.01, 0.07\rangle$
0.75	$\langle 0.01, 0.41, 0.00, 0.01, 0.10, 0.35, 0.01, 0.11\rangle$
1.00	$\langle 0.02, 0.45, 0.00, 0.01, 0.14, 0.23, 0.01, 0.15\rangle$

When the agent considers the observation completely epistemically, its beliefs change very similarly to when it was only told that the grass is wet; the agent already believed that the book was dry. However, the extra information has a significant impact on how the agent's beliefs change when the observation is considered ontically. The agent now believes much less that the book is outside and wet and the grass is wet, and with 78% (as opposed to 40% with the first observation) that the book is *dry* and the grass is wet (independent of where the book is located). The reason is that when the received information includes a dry book, transitions are focused on going to dry-book worlds.

[5] I shall assume that the null event may include some unknown events (with unknown effects).

Conclusion

In this paper I suggested a method to arrive at a new (probabilistic) belief state when the agent has mixed feelings about whether to revise or update its beliefs, given a new piece of information. Much attention was given to the design and analysis of the separate update and revision operations. The postulates and finally Theorem 1 add weight to my argument that the hybrid stochastic belief change (HSBC) operation is rational when the agent has a rational frame of mind.

Looking at the examples above, the way in which probabilities shift among the possible worlds, given the different ontic/epistemic strengths, seems justifiable. However, more analysis is required here, especially when considering more complicated specification patterns of the ontic/epistemic strengths.

Determining $os(\alpha, w)$ for every foreseen α in every possible world w will be challenging for a designer. Some deep questions are: Should the designer/agent provide the strengths (via stored values or programmed reasoning), or do these strengths come to the agent *attached* to the new information? What is the reasoning process we go through to determine whether information is epistemic or ontic, if at all? In general, how does an agent know when information is epistemic (requiring revision) or ontic (requiring update)?

References

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50(2):510–530.

Aström, K. 1965. Optimal control of Markov decision processes with incomplete state estimation. *Journal of Mathematical Analysis and Applications* 10:174–205.

Boutilier, C. 1994. Unifying default reasoning and belief revision in a modal framework. *Artificial Intelligence* 68:33–85.

Boutilier, C. 1998. A unified model of qualitative belief change: a dynamical systems perspective. *Artificial Intelligence* 98(1–2):281–316.

Chhogyal, K.; Nayak, A.; Schwitter, R.; and Sattar, A. 2014. Proceedings of the thirteenth pacific rim international conference on artificial intelligence (pricai 2014). In Pham, D., and Park, S., eds., *Proc. of PRICAI 2014*, volume 8862 of *LNCS*, 694–707. Springer-Verlag.

Dubois, D., and Prade, H. 1993. Belief revision and updates in numerical formalisms: An overview, with new results for the possibilistic framework. In *Proceedings of the 13th International Joint Conference on Artifical Intelligence*, volume 1 of *IJCAI'93*, 620–625. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Gärdenfors, P. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Massachusetts/England: MIT Press.

Goldszmidt, M., and Pearl, J. 1992. Rank-based systems: A simple approach to belief revision, belief update, and reasoning about evidence and actions. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, 661–672. Cambridge.

Grove, A., and Halpern, J. 1998. Updating sets of probabilities. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, 173–182. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Grove, A. 1988. Two modellings for theory change. *Journal of Philosophical Logic* 17:157–170.

Katsuno, H., and Mendelzon, A. 1991. On the difference between updating a knowledge base and revising it. In *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, 387–394.

Kern-Isberner, G. 2008. Linking iterated belief change operations to nonmonotonic reasoning. In *Proceedings of the Eleventh International Conference on Principles of Knowledge Representation and Reasoning*, 166–176. Menlo Park, CA: AAAI Press.

Koller, D., and Friedman, N. 2009. *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA and London, England: The MIT Press.

Lewis, D. 1976. Probabilities of conditionals and conditional probabilities. *Philosophical Review* 85(3):297–315.

Lovejoy, W. 1991. A survey of algorithmic methods for partially observed Markov decision processes. *Annals of Operations Research* 28:47–66.

Monahan, G. 1982. A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science* 28(1):1–16.

Moore, R. 1990. A formal theory of knowledge and action. In Allen, J.; Hendler, J.; and Tate, A., eds., *Readings in Planning*. San Mateo: Morgan-Kaufmann. 480–519.

Poole, D., and Mackworth, A. 2010. *Artificial Intelligence: Foundations of Computational Agents*. New York, USA: Cambridge University Press.

Rens, G., and Meyer, T. 2015. A new approach to probabilistic belief change. In Russell, I., and Eberle, W., eds., *Proceedings of the International Florida AI Research Society Conference (FLAIRS)*, 582–587. AAAI Press.

Spohn, W. 1988. Ordinal conditional functions: A dynamic theory of epistemic states. In Harper, W., and Skyrms, B., eds., *Causation in Decision, Belief Change, and Statistics*, volume 42 of *The University of Western Ontario Series in Philosophy of Science*. Springer Netherlands. 105–134.

Voorbraak, F. 1999. Partial Probability: Theory and Applications. In *Proceedings of the First International Symposium on Imprecise Probabilities and Their Applications*, 360–368. url: decsai.ugr.es/ smc/isipta99/proc/073.html.

Yue, A., and Liu, W. 2008. Revising imprecise probabilistic beliefs in the framework of probabilistic logic programming. In *Proceedings of the Twenty-third AAAI Conf. on Artificial Intelligence (AAAI-08)*, 590–596.

Revising Incompletely Specified Convex Probabilistic Belief Bases

Gavin Rens
CAIR[*],
University of KwaZulu-Natal,
School of Mathematics, Statistics and Comp. Sci.
CSIR Meraka, South Africa
Email: gavinrens@gmail.com

Thomas Meyer
CAIR,
University of Cape Town,
Dept. of Comp. Sci.
CSIR Meraka, South Africa
Email: tmeyer@cs.uct.ac.za

Giovanni Casini
University of Luxembourg,
Comp. Sci. and Communication Research Unit
Luxembourg
Email: giovanni.casini@uni.lu

Abstract

We propose a method for an agent to revise its incomplete probabilistic beliefs when a new piece of propositional information is observed. In this work, an agent's beliefs are represented by a set of probabilistic formulae – a belief base. The method involves determining a *representative* set of 'boundary' probability distributions consistent with the current belief base, revising each of these probability distributions and then translating the revised information into a new belief base. We use a version of Lewis Imaging as the revision operation. The correctness of the approach is proved. The expressivity of the belief bases under consideration are rather restricted, but has some applications. We also discuss methods of belief base revision employing the notion of optimum entropy, and point out some of the benefits and difficulties in those methods. Both the boundary distribution method and the optimum entropy method are reasonable, yet yield different results.

Suppose an agent represents its probabilistic knowledge with a set of statements; every statement says something about the probability of some features the agent is aware of. Ideally, the agent would want to have enough information to, at least, identify one probability distribution over all the situations (worlds) it deems possible. However, if the agent could not gather sufficient data or if it was not told or given sufficient information, it would not be able to pinpoint exactly one probability distribution. An agent with this sort of ignorance, can be thought of as having beliefs compatible with a *set of* distributions. Now, this agent might need to revise its beliefs when new (non-probabilistic) information is received, even though the agent's beliefs do not characterize a *particular* probability distribution over its current possible worlds.

Several researchers argue that using a single probability distribution requires the agent to make unrealistically precise uncertainty distinctions (Grove and Halpern, 1998; Voorbraak, 1999; Yue and Liu, 2008).[1] "One widely-used approach to dealing with this has been to consider sets of probability measures as a way of modeling uncertainty," (Grove and Halpern, 1998). However, simply applying standard probabilistic conditioning to each of the measures/distributions in the set individually and then combining the results is also not recommended. The framework presented in this paper proposes two ways to go from one 'probabilistically incomplete' belief base to another when new information is acquired.

Both belief revision methods presented, essentially follow this process: From the original belief base, determine a relatively small set of belief states / probability distributions 'compatible' with the belief base which is, in a sense, representative of the belief base. (We shall use the terms *belief state*, *probability distribution*, *probability function* and *distribution* interchangeably). Then revise every belief state in this representative set. Finally, induce a new, revised belief base from the revised representative set.

We shall present two approaches to determine the representative set of belief states from the current belief base: (i) The approach we focus on involves finding belief states which, in a sense, are at the boundaries of the constraints implied by the belief base. These 'boundary belief states' can be thought of as drawing the outline of the convex space of beliefs. This outline is then revised to form a new outline shape, which can be translated into a new belief base. (ii) As a possible alternative approach, the representative set is a *single* belief state which can be imagined to be at the center of the outline of the first approach. This 'central' belief state is found by determining the one in the space of beliefs which is least biased or most entropic in terms of information theory (Jaynes, 1978; Cover and Thomas, 1991).

For approach (i) – where the canonical set is the set of boundary belief states – we shall prove that the revised canonical set characterizes the set of all belief states which would have resulted from revising all (including interior) belief states compatible with the original belief base.

The relevant background theory and notations are now introduced.

We shall work with classical propositional logic. Let $\mathcal{P}$ be the finite set of atomic propositional variables (*atoms*, for short). Formally, a *world* is a unique assignment of truth values to all the atoms in $\mathcal{P}$. There are thus 2^n conceivable worlds. An agent may consider some non-empty subset W

[*] Centre for Artificial Intelligence Research

[1] See also the references in these cited papers concerning criticisms against traditional probability theory.

of the conceivable worlds called the possible worlds. Often, in the exposition of this paper, a world will be referred to by its truth vector. For instance, if the vocabulary is placed in order $\langle q, r\rangle$ and $w_3 \Vdash \neg q \wedge r$, then w_3 may be referred to as 01.[2] Let L be all propositional formulae which can be formed from $\mathcal{P}$ and the logical connectives $\wedge$ and $\neg$, with $\top$ abbreviating tautology and $\bot$ abbreviating contradiction.

Let β be a sentence in L. $[\beta]$ denotes the set of β-worlds, that is, the elements of W satisfying β. The worlds satisfying all sentences in a set of sentences K are denoted by $[K]$.

We define the probabilistic language $L^{prob} = \{(\alpha) \bowtie x \mid \alpha \in L, \bowtie \in \{\leq, =, \geq\}, x \in [0,1]\}$. Sentences with strict inequalities ($<, >$) are excluded from the language for now. Such sentences are more challenging to deal with and their inclusion is left for future work. We propose a belief base (BB) to be a consistent (logically satisfiable) subset of L^{prob}. A BB specifies an agent's knowledge.

The basic semantic element of an agent's beliefs is a probability distribution or a *belief state*

$$b = \{(w_1, p_1), (w_2, p_2), \ldots, (w_n, p_n)\},$$

where p_i is the probability that w_i is the actual world in which the agent is. $\sum_{(w,p)\in b} p = 1$. We may also use c to refer to a belief state. For parsimony, let $b = \langle p_1, \ldots, p_n\rangle$ be the probabilities that belief state b assigns to $w_1, \ldots, w_n$ where $\langle w_1, w_2, w_3, w_4\rangle = \langle 11, 10, 01, 00\rangle$, and $\langle w_1, w_2, \ldots, w_8\rangle = \langle 111, 110, \ldots, 000\rangle$. Let Π be the set of all belief states over W.

$b(\alpha)$ abbreviates $\sum_{w\in W, w\Vdash\alpha} b(w)$. b *satisfies* formula $(\alpha) \bowtie x$ (denoted $b \Vdash (\alpha) \bowtie x$) iff $b(\alpha) \bowtie x$. If B is a set of formulae, then b *satisfies* B (denoted $b \Vdash B$) iff $\forall\gamma \in B$, $b \Vdash \gamma$. If B and B' are sets of formulae, then B *entails* B' (denoted $B \models B'$) iff for all $b \in \Pi$, $b \Vdash B'$ whenever $b \Vdash B$. If $B \models \{\gamma\}$ then we simply write $B \models \gamma$. B is logically equivalent to B' (denoted $B \equiv B'$) iff $B \models B'$ and $B' \models B$.

Instead of an agent's beliefs being represented by a single belief state, a BB B represents a *set* of belief-states: Let $\Pi^B := \{b \in \Pi \mid b \Vdash B\}$. A BB B is *satisfiable* (*consistent*) iff $\Pi^B \neq \emptyset$.

The technique of *Lewis imaging* for the revision of belief states, requires a notion of distance between worlds to be defined. We use a pseudo-distance measure between worlds, as defined by Lehmann, Magidor, and Schlechta (2001) and adopted by Chhogyal et al. (2014).

We add a 'faithfulness' condition, which we feel is lacking from the definition of Lehmann, Magidor, and Schlechta (2001): without this condition, a pseudo-distance measure would allow all worlds to have zero distance between them. Boutilier (1998) mentions this condition, and we use his terminology: "faithfulness".

Definition 1. *A pseudo-distance function* $d : W \times W \to \mathbb{Z}$ *satisfies the following four conditions: for all worlds* $w, w', w'' \in W$,

1. $d(w, w') \geq 0$ *(Non-negativity)*
2. $d(w, w) = 0$ *(Identity)*
3. $d(w, w') = d(w', w)$ *(Symmetry)*
4. $d(w, w') + d(w', w'') \geq d(w, w'')$ *(Triangular Inequality)*
5. *if* $w \neq w'$*, then* $d(w, w') > 0$ *(Faithfulness)*

Presently, the foundation theory, or paradigm, for studying belief change operations is commonly known as AGM theory (Alchourrón, Gärdenfors, and Makinson, 1985; Gärdenfors, 1988). Typically, belief change (in a static world) can be categorized as expansion, revision or contraction, and is performed on a belief set, the set of sentences K closed under logical consequence. Expansion (denoted $+$) is the logical consequences of $K \cup \{\alpha\}$, where α is new information and K is the current belief set. Contraction of α is the removal of some sentences until α cannot be inferred from K. It is the reduction of beliefs. Revision is when α is (possibly) inconsistent with K and K is (minimally) modified so that the new K remains consistent and entails α. In this view, when the new information is consistent with the original beliefs, expansion and revision are equivalent.

The next section presents a generalized imaging method for revising probabilistic belief states. Then we describe the application of generalized imaging in our main contribution; revising boundary belief states instead of all belief states. The subsequent section explain another approaches of revising our belief bases, which prepares us for discussions in the rest of the paper. The latter method finds a single representative belief state through maximum entropy inference. Both the boundary belief state method and the maximum entropy method are reasonable, yet yield different results – a seeming paradox is thus uncovered. Then future possible directions of research are discussed. We end with a section on the related work and the concluding section.

Generalized Imaging

It is not yet universally agreed what revision means in a probabilistic setting. One school of thought says that probabilistic expansion is equivalent to Bayesian conditioning. This is evidenced by Bayesian conditioning (BC) being defined only when $b(\alpha) \neq 0$, thus making BC expansion equivalent to BC revision. In other words, one could define expansion (restricted revision) to be

$$b\,\mathsf{BC}\,\alpha = \{(w, p) \mid w \in W, p = b(w \mid \alpha), b(\alpha) \neq 0\}.$$

To accommodate cases where $b(\alpha) = 0$, that is, where α contradicts the agent's current beliefs and its beliefs need to be revised in the stronger sense, we shall make use of *imaging*. Imaging was introduced by Lewis (1976) as a means of revising a probability function. It has also been discussed in the work of, for instance, Gärdenfors (1988); Dubois and Prade (1993); Chhogyal et al. (2014); Rens and Meyer (2015). Informally, Lewis's original solution for accommodating contradicting evidence α is to move the probability of each world to its closest, α-world. Lewis made the strong assumption that every world has a *unique* closest α-world. More general versions of imaging allows worlds to have *several*, equally proximate, closest worlds.

[2] $w \Vdash \alpha$ is read 'w is a model for/satisfies α'.

Gärdenfors (1988) calls one of his generalizations of Lewis's imaging *general imaging*. Our method is also a generalization. We thus refer to his as *Gärdenfors's general imaging* and to our method as *generalized imaging* to distinguish them. It should be noted that all three these imaging methods are general revision methods and can be used in place of Bayesian conditioning for expansion. "Thus imaging is a more general method of describing belief changes than conditionalization," (Gärdenfors, 1988, p. 112).

Let $Min(\alpha, w, d)$ be the set of α-worlds closest to w with respect to pseudo-distance d. Formally,

$$\begin{aligned}&Min(\alpha, w, d) := \\ &\quad \{w' \in [\alpha] \mid \forall w'' \in [\alpha], d(w', w) \leq d(w'', w)\},\end{aligned}$$

where $d(\cdot)$ is some pseudo-distance measure between worlds (e.g., Hamming or Dalal distance).

Example 1. *Let the vocabulary be* $\{q, r, s\}$*. Let* α *be* $(q \wedge r) \vee (q \wedge \neg r \wedge s)$*. Suppose* d *is Hamming distance. Then*

$$\begin{aligned}
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 111, d) = \{111\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 110, d) = \{110\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 101, d) = \{101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 100, d) = \{110, 101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 011, d) = \{111\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 010, d) = \{110\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 001, d) = \{101\}\\
&Min((q \wedge r) \vee (q \wedge \neg r \wedge s), 000, d) = \{110, 101\}
\end{aligned}$$

□

Definition 2 (GI)**.** *Then* generalized imaging *(denoted* GI*) is defined as*

$$\begin{aligned}&b \,\mathsf{GI}\, \alpha := \{(w, p) \mid w \in W, p = 0 \textit{ if } w \notin [\alpha],\\ &\qquad \textit{else } p = \sum_{\substack{w' \in W\\ w \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)|\}.\end{aligned}$$

In words, $b \,\mathsf{GI}\, \alpha$ is the new belief state produced by taking the generalized image of b with respect to α. Notice how the probability mass of non-α-worlds is shifted to their closest α-worlds. If a non-α-world $w^{\times}$ with probability p has n closest α-worlds (equally distant), then each of these closest α-worlds gets p/n mass from $w^{\times}$.

We define $b^{\circ}_{\alpha} := b \circ \alpha$ so that we can write $b^{\circ}_{\alpha}(w)$, where $\circ$ is a revision operator.

Example 2. *Continuing on Example 1: Let* $b = \langle 0, 0.1, 0, 0.2, 0, 0.3, 0, 0.4\rangle$*.*

$(q \wedge r) \vee (q \wedge \neg r \wedge s)$ *is abbreviated as* α*.*

$b^{\mathsf{GI}}_{\alpha}(111) = \sum_{\substack{w' \in W\\ 111 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(111)/|Min(\alpha, 111, d)| + b(011)/|Min(\alpha, 011, d)| = 0/1 + 0/1 = 0.$

$b^{\mathsf{GI}}_{\alpha}(110) = \sum_{\substack{w' \in W\\ 110 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(110)/|Min(\alpha, 110, d)| + b(100)/|Min(\alpha, 100, d)| + b(010)/|Min(\alpha, 010, d)| + b(000)/|Min(\alpha, 000, d)| = 0.1/1 + 0.2/2 + 0.3/1 + 0.4/2 = 0.7.$

$b^{\mathsf{GI}}_{\alpha}(101) = \sum_{\substack{w' \in W\\ 101 \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)| = b(101)/|Min(\alpha, 101, d)| + b(100)/|Min(\alpha, 100, d)| + b(001)/|Min(\alpha, 001, d)| + b(000)/|Min(\alpha, 000, d)| = 0/1 + 0.2/2 + 0/1 + 0.4/2 = 0.3.$

And $b^{\mathsf{GI}}_{\alpha}(100) = b^{\mathsf{GI}}_{\alpha}(011) = b^{\mathsf{GI}}_{\alpha}(010) = b^{\mathsf{GI}}_{\alpha}(001) = b^{\mathsf{GI}}_{\alpha}(000) = 0.$ □

Revision via GI and boundary belief states

Perhaps the most obvious way to revise a given belief base (BB) B is to revise every individual belief state in Π^B and then induce a new BB from the set of revised belief states. Formally, given observation α, first determine a new belief state b^{α} for every $b \in \Pi^B$ via the defined revision operation:

$$\Pi^{B^{\alpha}} = \{b^{\alpha} \in \Pi \mid b^{\alpha} = b \,\mathsf{GI}\, \alpha,\ b \in \Pi^B\}.$$

If there is more than only a single belief state in Π^B, then Π^B contains an infinite number of belief states. Then how can one compute $\Pi^{B^{\alpha}}$? And how would one subsequently determine B^{α} from $\Pi^{B^{\alpha}}$?

In the rest of this section we shall present a finite method of determining $\Pi^{B^{\alpha}}$. What makes this method possible is the insight that Π^B can be represented by a finite set of 'boundary' belief states – those belief states which, in a sense, represent the limits or the convex hull of Π^B. We shall prove that the set of revised boundary belief states defines $\Pi^{B^{\alpha}}$. Inducing B^{α} from $\Pi^{B^{\alpha}}$ is then relatively easy, as will be seen.

Let W^{perm} be every permutation on the ordering of worlds in W. For instance, if $W = \{w_1, w_2, w_3, w_4\}$, then $W^{perm} = \{\langle w_1, w_2, w_3, w_4\rangle, \langle w_1, w_2, w_4, w_3\rangle, \langle w_1, w_3, w_2, w_4\rangle, \ldots, \langle w_4, w_3, w_2, w_1\rangle\}$. Given an ordering $W^{\#} \in W^{perm}$, let $W^{\#}(i)$ be the i-th element of $W^{\#}$; for instance, $\langle w_4, w_3, w_2, w_1\rangle(2) = w_3$. Suppose we are given a BB B. We now define a function which, given a permutation of worlds, returns a belief state where worlds earlier in the ordering are assigned maximal probabilities according to the boundary values enforced by B.

Definition 3. $MaxASAP(B, W^{\#})$ *is the* $b \in \Pi^B$ *such that for* $i = 1, \ldots, |W|$*,* $\forall b' \in \Pi^B$*, if* $b' \neq b$*, then* $\sum_{j=1}^{i} b(W^{\#}(j)) \geq \sum_{k=1}^{i} b'(W^{\#}(k))$*.*

Example 3. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_1 = \{(q) \geq 0.6\}$*. Then, for instance,* $MaxASAP(B_1, \langle 01, 00, 11, 10\rangle) = \{(01, 0.4), (00, 0), (11, 0.6), (10, 0)\} = \{(11, 0.6), (10, 0), (01, 0.4), (00, 0)\}$. □

Definition 4. *We define the boundary belief states of BB* B *as the set*

$$\begin{aligned}&\Pi^B_{bnd} := \{b \in \Pi^B \mid\\ &\qquad W^{\#} \in W^{perm}, b = MaxASAP(B, W^{\#})\}\end{aligned}$$

Note that $|\Pi^B_{bnd}| \leq |W^{perm}|$.

Example 4. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_1 = \{(q) \geq 0.6\}$. *Then*

$$\begin{array}{rcl} \Pi^{B_1}_{bnd} & = & \{\{(11, 1.0), (10, 0.0), (01, 0.0), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 1.0), (01, 0.0), (00, 0.0)\}, \\ & & \{(11, 0.6), (10, 0.0), (01, 0.4), (00, 0.0)\}, \\ & & \{(11, 0.6), (10, 0.0), (01, 0.0), (00, 0.4)\}, \\ & & \{(11, 0.0), (10, 0.6), (01, 0.4), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 0.6), (01, 0.0), (00, 0.4)\}\}. \end{array}$$

□

Next, the revision operation is applied to every belief state in Π^B_{bnd}. Let $(\Pi^B_{bnd})^{\mathsf{GI}}_\alpha := \{b' \in \Pi \mid b' = b^{\mathsf{GI}}_\alpha,\ b \in \Pi^B_{bnd}\}$.

Example 5. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_1 = \{(q) \geq 0.6\}$. *Let* α *be* $(q \wedge \neg r) \vee (\neg q \wedge r)$. *Then*

$$\begin{array}{rcl} (\Pi^{B_1}_{bnd})^{\mathsf{GI}}_\alpha & = & \{\{(11, 0.0), (10, 0.5), (01, 0.5), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 1.0), (01, 0.0), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 0.3), (01, 0.7), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 0.6), (01, 0.4), (00, 0.0)\}, \\ & & \{(11, 0.0), (10, 0.8), (01, 0.2), (00, 0.0)\}\}. \end{array}$$

(Two revision operations produce $\{(11, 0), (10, 0.5), (01, 0.5), (00, 0)\}$.*)* □

To induce the new BB B^α_{bnd} from $(\Pi^B_{bnd})^{\mathsf{GI}}_\alpha$, the following procedure is executed. For every possible world, the procedure adds a sentence enforcing the upper (resp., lower) probability limit of the world, with respect to all the revised boundary belief states. Trivial limits are excepted.

> For every $w \in W$, $(\phi_w) \leq \overline{y} \in B^\alpha$, where $\overline{y} = \max_{b \in (\Pi^B_{bnd})^{\mathsf{GI}}_\alpha} b(w)$, except when $\overline{y} = 1$, and $(\phi_w) \geq \underline{y} \in B^\alpha$, where $\underline{y} = \min_{b \in (\Pi^B_{bnd})^{\mathsf{GI}}_\alpha} b(w)$, except when $\underline{y} = 0$.

The intention is that the procedure specifies B^α to represent the upper and lower probability envelopes of the set of revised boundary belief states – B^α thus defines the entire revised belief state space (cf. Theorem 1).

Example 6. *Continuing Example 5, using the translation procedure just above, we see that* $B^\alpha_{1bnd} = \{(\phi_{11}) \leq 0$, $(\phi_{10}) \geq 0.3$, $(\phi_{01}) \leq 0.7$, $(\phi_{00}) \leq 0.0\}$.

Note that if we let $B' = \{((q \wedge \neg r) \vee (\neg q \wedge r)) = 1$, $(q \wedge \neg r) \geq 0.3\}$, *then* $\Pi^{B'} = \Pi^{B^\alpha_{1bnd}}$. □

Example 7. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_2 = \{(\neg q \wedge \neg r) = 0.1\}$. *Let* α *be* $\neg q$. *Then*

$$\begin{array}{rcl} \Pi^{B_2}_{bnd} & = & \{\{(11, 0.9), (10, 0), (01, 0), (00, 0.1)\}, \\ & & \{(11, 0), (10, 0.9), (01, 0), (00, 0.1)\}, \\ & & \{(11, 0), (10, 0), (01, 0.9), (00, 0.1)\}\}, \end{array}$$

$$\begin{array}{rcl} (\Pi^{B_2}_{bnd})^{\mathsf{GI}}_\alpha & = & \{\{(11, 0), (10, 0), (01, 0.9), (00, 0.1)\}, \\ & & \{(11, 0), (10, 0), (01, 0), (00, 1)\}\} \textit{ and} \end{array}$$

$B^\alpha_{2bnd} = \{(\phi_{11}) \leq 0$, $(\phi_{10}) \leq 0$, $(\phi_{01}) \leq 0.9$, $(\phi_{00}) \geq 0.1\}$.

Note that if we let $B' = \{(\neg q) = 1$, $(\neg q \wedge r) \leq 0.9\}$, *then* $\Pi^{B'} = \Pi^{B^\alpha_{2bnd}}$. □

Let $W^{Min(\alpha, d)}$ be a partition of W such that $\{w^i_1, \ldots, w^i_{ni}\}$ is a block in $W^{Min(\alpha, d)}$ iff $|Min(\alpha, w^i_1, d)| = \cdots = |Min(\alpha, w^i_{ni}, d)|$. Denote an element of block $\{w^i_1, \ldots, w^i_{ni}\}$ as w^i, and the block of which w^i is an element as $[w^i]$. Let $i = |Min(\alpha, w^i, d)|$, in other words, the superscript in w^i indicates the size of $Min(\alpha, w^i, d)$. Let $m := \max_{w \in W} |Min(\alpha, w, d)|$.

Observation 1. *Let* $\delta_1, \delta_2, \ldots, \delta_m$ *be positive integers such that* $i < j$ *iff* $\delta_i < \delta_j$. *Let* $\nu_1, \nu_2, \ldots, \nu_m$ *be values in* $[0, 1]$ *such that* $\sum_{k=1}^m \nu_k = 1$. *Associate with every* ν_i *a maximum value it is allowed to take:* $most(\nu_i)$. *For every* ν_i, *we define the* assignment value

$$av(\nu_i) := \begin{cases} most(\nu_i) & \textit{if } \sum_{k=1}^i \leq 1 \\ 1 - \sum_{k=1}^{i-1} & \textit{otherwise} \end{cases}$$

Determine first $av(\nu_1)$, *then* $av(\nu_2)$ *and so on. Then*

$$\frac{av(\nu_1)}{\delta_1} + \cdots + \frac{av(\nu_m)}{\delta_m} > \frac{\nu'_1}{\delta_1} + \cdots + \frac{\nu'_m}{\delta_m}$$

whenever $\nu'_i \neq av(\nu_i)$ *for some* i. □

For instance, let $\delta_1 = 1$, $\delta_2 = 2$, $\delta_3 = 3$, $\delta_4 = 4$. Let $most(\nu_1) = 0.5$, $most(\nu_2) = 0.3$, $most(\nu_3) = 0.2$, $most(\nu_4) = 0.3$. Then $av(\nu_1) = 0.5$, $av(\nu_2) = 0.3$, $av(\nu_3) = 0.2$, $av(\nu_4) = 0$ and

$$\frac{0.5}{1} + \frac{0.3}{2} + \frac{0.2}{3} + \frac{0}{4} = 0.716.$$

But

$$\frac{0.49}{1} + \frac{0.3}{2} + \frac{0.2}{3} + \frac{0.01}{4} = 0.709.$$

And

$$\frac{0.5}{1} + \frac{0.29}{2} + \frac{0.2}{3} + \frac{0.01}{4} = 0.714.$$

Lemma 1 essentially says that the belief state in Π^B which causes a revised belief state to have a maximal value at world w (w.r.t. all belief states in Π^B), will be in Π^B_{bnd}.

Lemma 1. *For all* $w \in W$, $\arg\max_{b_X \in \Pi^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_X(w')/|Min(\alpha, w', d)|$ *is in* Π^B_{bnd}.

Proof. Note that

$$\sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)|$$

can be written in the form

$$\frac{\sum_{\substack{w' \in [w^1] \\ w \in Min(\alpha, w', d)}} b(w')}{1} + \cdots + \frac{\sum_{\substack{w' \in [w^m] \\ w \in Min(\alpha, w', d)}} b(w')}{m}.$$

Observe that there must be a $W^\# \in W^{perm}$ such that $W^\# = \langle w^1_1, \ldots, w^1_{n1}, \ldots, w^m_1, \ldots, w^m_{nm} \rangle$. Then by the definition of the set of boundary belief states (Def. 4), $MaxASAP(B, W^{\#})$ will assign maximal probability mass to $[w^1] = \{w_1^1, \ldots, w_{n1}^1\}$, then to $[w^2] = \{w_1^2, \ldots, w_{n2}^m\}$ and so on.

That is, by Observation 1, for some $b_x \in \Pi_{bnd}^B$, $b_x(w) = max_{b_X \in \Pi^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_X(w')/|Min(\alpha, w', d)|$ for all $w \in W$. Therefore, $\arg\max_{b_X \in \Pi^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_X(w')/|Min(\alpha, w', d)|$ is in Π_{bnd}^B. □

Let

$\overline{x}^w := \max_{b \in \Pi_{bnd}^B} b(w)$ $\quad \overline{X}^w := \max_{b \in \Pi^B} b(w)$

$\overline{y}^w := \max_{b \in (\Pi_{bnd}^B)_\alpha^{GI}} b(w)$ $\quad \overline{Y}^w := \max_{b \in (\Pi^B)_\alpha^{GI}} b(w)$

$\underline{x}^w := \min_{b \in \Pi_{bnd}^B} b(w)$ $\quad \underline{X}^w := \min_{b \in \Pi^B} b(w)$

$\underline{y}^w := \min_{b \in (\Pi_{bnd}^B)_\alpha^{GI}} b(w)$ $\quad \underline{Y}^w := \min_{b \in (\Pi^B)_\alpha^{GI}} b(w)$

Lemma 2 states that for every world, the upper/lower probability of the world with respect to Π_{bnd}^B is equal to the upper/lower probability of the world with respect to Π^B. The proof requires Observation 1 and Lemma 1.

Lemma 2. *For all $w \in W$, $\overline{y}^w = \overline{Y}^w$ and $\underline{y}^w = \underline{Y}^w$.*

Proof. Note that if $w \notin [\alpha]$, then $\overline{y}^w = \overline{Y}^w = 0$ and $\underline{y}^w = \underline{Y}^w = 0$.

We now consider the cases where $w \in [\alpha]$.

$$\overline{y}^w = \overline{Y}^w$$

iff

$$\max_{b \in (\Pi_{bnd}^B)} b(w) = \max_{b \in (\Pi^B)} b(w)$$

iff

$$\max_{b_x \in \Pi_{bnd}^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_x(w')/|Min(\alpha, w', d)|$$

$$= \max_{b_X \in \Pi^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_X(w')/|Min(\alpha, w', d)|$$

if

$\bar{b}_x(w) = \bar{b}_X(w)$, where

$$\bar{b}_x(w) := \max_{b_x \in \Pi_{bnd}^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_x(w')/|Min(\alpha, w', d)|$$

and

$$\bar{b}_X(w) := \max_{b_X \in \Pi^B} \sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_X(w')/|Min(\alpha, w', d)|.$$

Note that

$$\sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b(w')/|Min(\alpha, w', d)|$$

can be written in the form

$$\frac{\sum_{\substack{w' \in [w^1] \\ w \in Min(\alpha, w', d)}} b(w')}{1} + \cdots + \frac{\sum_{\substack{w' \in [w^m] \\ w \in Min(\alpha, w', d)}} b(w')}{m}.$$

Then by Observation 1, $\bar{b}_X(w)$ is in Π_{bnd}^B. And also by Lemma 1, the belief state in Π_{bnd}^B identified by $\bar{b}_X(w)$ must be the one which maximizes

$$\sum_{\substack{w' \in W \\ w \in Min(\alpha, w', d)}} b_x(w')/|Min(\alpha, w', d)|,$$

where $b_x \in \Pi_{bnd}^B$. That is, $\bar{b}_x = \bar{b}_X$.

With a symmetrical argument, it can be shown that $\underline{y}^w = \underline{Y}^w$. □

In intuitive language, the following theorem says that the BB determined through the method of revising boundary belief states captures exactly the same beliefs and ignorance as the belief states in Π^B which have been revised. This correspondence relies on the fact that the upper and lower probability envelopes of Π^B can be induce from Π_{bnd}^B, which is what Lemma 2 states.

Theorem 1. *Let $(\Pi^B)_\alpha^{\mathsf{GI}} := \{b_\alpha^{\mathsf{GI}} \in \Pi \mid b \in \Pi^B\}$. Let B_{bnd}^α be the BB induced from $(\Pi_{bnd}^B)_\alpha^{\mathsf{GI}}$. Then $\Pi^{B_{bnd}^\alpha} = (\Pi^B)_\alpha^{\mathsf{GI}}$.*

Proof. We show that $\forall b' \in \Pi$, $b' \in \Pi^{B_{bnd}^\alpha} \iff b' \in (\Pi^B)_\alpha^{\mathsf{GI}}$.

($\Rightarrow$) $b' \in \Pi^{B_{bnd}^\alpha}$ implies $\forall w \in W$, $\underline{y}^w \leq b'(w) \leq \overline{y}^w$ (by definition of B_{bnd}^α). Lemma 2 states that for all $w \in W$, $\overline{y}^w = \overline{Y}^w$ and $\underline{y}^w = \underline{Y}^w$. Hence, $\forall w \in W, \underline{Y}^w \leq b'(w) \leq \overline{Y}^w$ Therefore, $b'(w) \in (\Pi^B)_\alpha^{\mathsf{GI}}$.

($\Leftarrow$) $b'(w) \in (\Pi^B)_\alpha^{\mathsf{GI}}$ implies $\forall w \in W, \underline{Y}^w \leq b'(w) \leq \overline{Y}^w$. Hence, by Lemma 2, $\forall w \in W$, $\underline{y}^w \leq b'(w) \leq \overline{y}^w$. Therefore, by definition of B_{bnd}^α, $b' \in \Pi^{B_{bnd}^\alpha}$. □

Revising via a Representative Belief State

Another approach to the revision of a belief base (BB) is to determine a representative of Π^B (call it b_{rep}), change the representative belief state via the the defined revision operation and then induce a new BB from the revised representative belief state. Selecting a representative probability function from a family of such functions is not new (Goldszmidt, Morris, and Pearl, 1990; Paris, 1994, e.g.). More formally, given observation α, first determine $b_{rep} \in \Pi^B$, then compute its revision b_{rep}^α, and finally induce B^α from b_{rep}^α.

We shall represent Π^B (and thus B) by the single 'least biased' belief state, that is, the belief state in Π^B with *highest entropy*:

Definition 5 (Shannon Entropy)**.**

$$H(b) := -\sum_{w \in W} b(w) \ln b(w),$$

where b is a belief state.

Definition 6 (Maximum Entropy)**.** *Traditionally, given some set of distributions* Π*, the most entropic distribution in* Π *is defined as*

$$b^H := \arg\max_{b \in \Pi} H(b).$$

Suppose $B_2 = \{(\neg q \wedge \neg r) = 0.1\}$. Then the belief state $b \in \Pi^{B_2}$ satisfying the constraints posed by B_2 for which $H(b)$ is maximized is $b_{rep} = b^H = \langle 0.3, 0.3, 0.3, 0.1\rangle$.

The above distribution can be found directly by applying the principle of maximum entropy: The true belief state is estimated to be the one consistent with known constraints, but is otherwise as unbiased as possible, or "Given no other knowledge, assume that everything is as random as possible. That is, the probabilities are distributed as uniformly as possible consistent with the available information," (Poole and Mackworth, 2010). Obviously world 00 must be assigned probability 0.1. And the remaining 0.9 probability mass should be uniformly spread across the other three worlds.

Applying GI to b_{rep} on evidence $\neg q$ results in $b^{\neg q}_{rep} = \langle 0, 0, 0.6, 0.4\rangle$.

Example 8. *Suppose the vocabulary is* $\{q, r\}$*,* $B_1 = \{(q) \geq 0.6\}$ *and* α *is* $(q \wedge \neg r) \vee (\neg q \wedge r)$*. Then* $b_{rep} = \arg\max_{b \in \Pi^{B_1}} H(b) = \langle 0.3, 0.3, 0.2, 0.2\rangle$*. Applying* GI *to* b_{rep} *on* α *results in* $b^{\alpha}_{rep} = \langle 0, 0.61, 0.39, 0\rangle$*.* b^{α}_{rep} *can be translated into* B^{α}_{1rep} *as* $\{(q \wedge \neg r) = 0.61, (\neg q \wedge r) = 0.39\}$*.* □

Still using $\alpha = (q \wedge \neg r) \vee (\neg q \wedge r)$, notice that $\Pi^{B^{\alpha}_{1rep}} \neq \Pi^{B^{\alpha}_{1bnd}}$. But how different *are* $B^{\alpha}_{1rep} = \{(q \wedge \neg r) = 0.61, (\neg q \wedge r) = 0.39\}$ and $B^{\alpha}_{1bnd} = \{(q \wedge r) \leq 0, (q \wedge \neg r) \geq 0.3, (\neg q \wedge r) \leq 0.7, (\neg q \wedge \neg r) \leq 0.0\}$? Perhaps one should ask, how different B^{α}_{1rep} is from the representative of B^{α}_{1bnd}: The least biased belief state satisfying B^{α}_{1bnd} is $\langle 0, 0.5, 0.5, 0\rangle$. That is, How different are $\langle 0, 0.61, 0.39, 0\rangle$ and $\langle 0, 0.5, 0.5, 0\rangle$?

In the case of B_2, we could compare $B^{\neg q}_{2bnd} = \{(\phi_{11}) \leq 0, (\phi_{10}) \leq 0, (\phi_{01}) \leq 0.9, (\phi_{00}) \geq 0.1\}$ with $b^{\neg q}_{rep} = \langle 0, 0, 0.6, 0.4\rangle$. Or if we take the least biased belief state satisfying $B^{\neg q}_{2bnd}$, we can compare $\langle 0, 0, 0.5, 0.5\rangle$ with $\langle 0, 0, 0.6, 0.4\rangle$.

It has been extensively argued (Jaynes, 1978; Shore and Johnson, 1980; Paris and Vencovsk, 1997) that maximum entropy is a reasonable inference mechanism, if not the most reasonable one (w.r.t. probability constraints). And in the sense that the boundary belief states method requires no compression / information loss, it also seems like a very reasonable inference mechanism for revising BBs as defined here. Resolving this misalignment in the results of the two methods is an obvious task for future research.

Future Directions

Some important aspects still missing from our framework are the representation of conditional probabilistic information such as is done in the work of Kern-Isberner, and the association of information with its level of entrenchment. On the latter point, when one talks about probabilities or likelihoods, if one were to take a frequentist perspective, information observed more (less) often should become more (less) entrenched. Or, without considering observation frequencies, an agent could be designed to have, say, one or two sets of deeply entrenched background knowledge (e.g., domain constraints) which does not change or is more immune to change than 'regular' knowledge.

Given that we have found that the belief base resulting from revising via the boundary-belief-states approach differs from the belief base resulting from revising via the representative-belief-state approach, the question arises, When is it appropriate to use a representative belief state defined as the most entropic belief state of a given set Π^B? This is an important question, especially due to the popularity of employing the Maximum Entropy principle in cases of undespecified probabilistic knowledge (Jaynes, 1978; Goldszmidt, Morris, and Pearl, 1990; Hunter, 1991; Voorbraak, 1999; Kern-Isberner, 2001; Kern-Isberner and Rdder, 2004) and the principle's well-behavedness (Shore and Johnson, 1980; Paris, 1994; Kern-Isberner, 1998).

Katsuno and Mendelzon (1991) modified the eight AGM belief revision postulates (Alchourrón, Gärdenfors, and Makinson, 1985) to the following six (written in the notation of this paper), where $*$ is some revision operator.[3]

- $B^{\alpha}_{*} \models (\alpha) = 1$.
- If $B \cup \{(\alpha) = 1\}$ is satisfiable, then $B^{\alpha}_{*} \equiv B \cup \{(\alpha) = 1\}$.
- If $(\alpha) = 1$ is satisfiable, then B^{α}_{*} is also satisfiable.
- If $\alpha \equiv \beta$, then $B^{\alpha}_{*} \equiv B^{\beta}_{*}$.
- $B^{\alpha}_{*} \cup \{(\beta) = 1\} \models B^{\alpha \wedge \beta}_{*}$.
- If $B^{\alpha}_{*} \cup \{(\beta) = 1\}$ is satisfiable, then $B^{\alpha \wedge \beta}_{*} \models B^{\alpha}_{*} \cup \{(\beta) = 1\}$.

Testing the various revision operations against these postulates is left for a sequel paper.

An extended version of maximum entropy is *minimum cross-entropy* (MCE) (Kullback, 1968; Csiszár, 1975):

Definition 7 (Minimum Cross-Entropy)**.** *The 'directed divergence' of distribution c from distribution b is defined as*

$$R(c, b) := \sum_{w \in W} c(w) \ln \frac{c(w)}{b(w)}.$$

$R(c, b)$ *is undefined when* $b(w) = 0$ *while* $c(w) > 0$*; when* $c(w) = 0$*,* $R(c, b) = 0$*, because* $\lim_{x \to 0} \ln(x) = 0$*. Given new evidence* $\phi \in L^{prob}$*, the distribution c satisfying* ϕ *diverging least from current belief state b is*

$$\arg\min_{c \in \Pi, c \Vdash \phi} R(c, b).$$

Definition 8 (MCI)**.** *Then MCE inference (denoted (*MCI*)) is defined as*

$$b \,\mathsf{MCI}\, \alpha := \arg\min_{b' \in \Pi, b' \Vdash (\alpha) = 1} R(b', b).$$

In the following example, we interpret revision as MCE inference.

[3] In these postulates, it is sometimes necessary to write an observation α as a BB, i.e., as $\{(\alpha) = 1\}$ – in the present framework, observations are regarded as certain.

Example 9. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_1 = \{(q) \geq 0.6\}$. *Let* α *be* $(q \wedge \neg r) \vee (\neg q \wedge r)$. *Then*

$$\begin{aligned}\Pi^{B_1}_{bnd} \;=\; & \{\{(11, 1.0), (10, 0.0), (01, 0.0), (00, 0.0)\},\\ & \{(11, 0.0), (10, 1.0), (01, 0.0), (00, 0.0)\},\\ & \{(11, 0.6), (10, 0.0), (01, 0.4), (00, 0.0)\},\\ & \{(11, 0.6), (10, 0.0), (01, 0.0), (00, 0.4)\},\\ & \{(11, 0.0), (10, 0.6), (01, 0.4), (00, 0.0)\},\\ & \{(11, 0.0), (10, 0.6), (01, 0.0), (00, 0.4)\}\},\end{aligned}$$

$$\begin{aligned}(\Pi^{B_1}_{bnd})^{\mathsf{MCI}}_{\alpha} \;=\; & \{\{(11, 0), (10, 0), (01, 1), (00, 0)\},\\ & \{(11, 0), (10, 1), (01, 0), (00, 0)\},\\ & \{(11, 0), (10, 0.6), (01, 0.4), (00, 0)\}\} \textit{ and}\end{aligned}$$

$B^{\alpha}_{1bnd} = \{(\phi_{11}) \leq 0, (\phi_{00}) \leq 0\}$.

Note that if we let $B' = \{((q \wedge \neg r) \vee (\neg q \wedge r)) = 1\}$, *then* $\Pi^{B'} = \Pi^{B^{\alpha}_{1bnd}}$. □

Recall from Example 6 that B' included $(q \wedge \neg r) \geq 0.3$. Hence, in this particular case, combining the boundary belief states approach with MCI results in a less informative revised belief base than when GI is used. The reason for the loss of information might be due to $R(\cdot, \{(11, 1.0), (10, 0.0), (01, 0.0), (00, 0.0)\})$ and $R(\cdot, \{(11, 0.6), (10, 0.0), (01, 0.0), (00, 0.4)\})$ being undefined: Recall that $R(c, b)$ is undefined when $b(w) = 0$ while $c(w) > 0$. But then there is no belief state c for which $c \Vdash \alpha$ and $R(\cdot)$ is defined (with these two belief states as arguments). Hence, there are no revised counterparts of these two belief states in $(\Pi^{B_1}_{bnd})^{\mathsf{MCI}}_{\alpha}$. We would like to analyse MCI more within this framework. In particular, in the future, we would like to determine whether a statement like Theorem 1 holds for MCI too.

In MCE inference, b-consistency of evidence ϕ is defined as: There exists a belief state c such that $c \Vdash \phi$ and c is *totally continuous* with respect to b (i.e., $b(w) = 0$ implies $c(w) = 0$). MCE is undefined when the evidence is not b-consistent. This is analogous to Bayesian conditioning being undefined for $b(\alpha) = 0$. Obviously, this is a limitation of MCE because some belief states may not be considered as candidate revised belief states. Admittedly, we have not searched the literature on this topic due to it being out of the present scope.

As far as we know, imaging for belief change has never been applied to (conditional) probabilistic evidence. Due to issues with many revision methods required to be consistent with prior beliefs, and imaging not having this limitation, it might be worthwhile investigating.

The translation from the set of belief states back to a belief base is a mapping from every belief state to a probability formula. The size of the belief base is thus in the order of $|W^{perm}|$, where $|W|$ is already exponential in the size of $\mathcal{P}$, the set of atoms. As we saw in several examples in this paper, the new belief base often has a more concise equivalent counterpart. It would be useful to find a way to consistently determine more concise belief bases than our present approach does.

The computational complexity of the process to revise a belief base is at least exponential. This work focused on theoretical issues. If the framework presented here is ever used in practice, computations will have to be optimized.

The following example illustrates how one might deal with strict inequalities.

Example 10. *Suppose the vocabulary is* $\{q, r\}$ *and* $B_3 = \{(q) > 0.6\}$. *Let* α *be* $(q \wedge \neg r) \vee (\neg q \wedge r)$. *Let* ϵ *be a real number which tends to 0. Then* $\Pi^{B_3}_{bnd} =$

$$\begin{aligned}& \{\{(11, 1.0), (10, 0.0), (01, 0.0), (00, 0.0)\},\\ & \{(11, 0.0), (10, 1.0), (01, 0.0), (00, 0.0)\},\\ & \{(11, 0.6 + \epsilon), (10, 0.0), (01, 0.4 - \epsilon), (00, 0.0)\},\\ & \{(11, 0.6 + \epsilon), (10, 0.0), (01, 0.0), (00, 0.4 - \epsilon)\},\\ & \{(11, 0.0), (10, 0.6 + \epsilon), (01, 0.4 - \epsilon), (00, 0.0)\},\\ & \{(11, 0.0), (10, 0.6 + \epsilon), (01, 0.0), (00, 0.4 - \epsilon)\}\},\end{aligned}$$

$(\Pi^{B_3}_{bnd})^{\mathsf{GI}}_{\alpha} =$

$$\begin{aligned}& \{\{(11, 0.0), (10, 0.5), (01, 0.5), (00, 0.0)\},\\ & \{(11, 0.0), (10, 1.0), (01, 0.0), (00, 0.0)\},\\ & \{(11, 0.0), (10, 0.3 + \epsilon), (01, 0.7 - \epsilon), (00, 0.0)\},\\ & \{(11, 0.0), (10, 0.6 + \epsilon), (01, 0.4 - \epsilon), (00, 0.0)\},\\ & \{(11, 0.0), (10, 0.8 + \epsilon), (01, 0.2 - \epsilon), (00, 0.0)\} \textit{ and}\end{aligned}$$

$B^{\alpha}_{3bnd} = \{(\phi_{11}) \leq 0, (\phi_{10}) \geq 0.3 + \epsilon, (\phi_{01}) \leq 0.7 - \epsilon, (\phi_{00}) \leq 0.0\}$.

Note that if we let $B' = \{((q \wedge \neg r) \vee (\neg q \wedge r)) = 1, (q \wedge \neg r) > 0.3\}$, *then* $\Pi^{B'} = \Pi^{B^{\alpha}_{3bnd}}$. □

It has been suggested by one of the reviewers that GI could be an *affine map* (i.t.o. geometry), thus allowing the proof of Theorem 1 to refer to existing results in the study of affine maps to significantly simplify the proof. The authors are not familiar with affine maps and thus leave investigation of the suggestion to other researchers.

Related Work

Voorbraak (1999) proposed the partial probability theory (PTT), which allows probability assignments to be partially determined, and where there is a distinction between probabilistic information based on (i) hard background evidence and (ii) some assumptions. He does not explicitly define the "constraint language", however, from his examples and discussions, one can infer that he has something like the language L^{PTT} in mind: it contains all formulae which can be formed with sentences in our L^{prob} in combination with connectives $\neg, \wedge$ and $\vee$. A "belief state" in PTT is defined as the quadruple $\langle \Omega, \mathcal{B}, \mathcal{A}, \mathcal{C} \rangle$, where Ω is a sample space, $\mathcal{B} \subset L^{PTT}$ is a sets of probability constraints, $\mathcal{A} \subset L^{PTT}$ is a sets of assumptions and $\mathcal{C} \subseteq W$ "represents specific information concerning the case at hand" (an observation or evidence).[4] Our epistemic state can be expressed as a restricted PTT "belief state" by letting $\Omega = W$, $\mathcal{B} = B$, $\mathcal{A} = \emptyset$ and

[4] Voorbraak (1999)'s "belief state" would rather be called and *epistemic state* or *knowledge structure* in our language.

$\mathcal{C} = \{w \in W \mid w \Vdash \alpha\}$, where B is a belief base and α is an observation in our notation.

Voorbraak (1999) mentions that he will only consider conditioning where the evidence does not contradict the current beliefs. He defines the set of belief states corresponding to the conditionalized PPT "belief state" as $\{b(\cdot \mid C) \in \Pi \mid b \in \Pi^{\mathcal{B} \cup \mathcal{A}}, b(C) > 0\}$. In our notation, this corresponds to $\{(b \,\mathsf{BC}\, \alpha) \in \Pi \mid b \in \Pi^{B}, b(\alpha) > 0\}$, where α corresponds to C.

Voorbraak (1999) proposes *constraining* as an alternative to conditioning: Let $\phi \in L^{prob}$ be a probability constraint. In our notation, constraining Π^{B} on ϕ produces $\Pi^{B \cup \{\phi\}}$.

Note that expanding a belief set reduces the number of models (worlds) and expanding a PPT "belief state" with extra constraints also reduces the number of models (belief states / probability functions).

> In the context of belief sets, it is possible to obtain any belief state from the ignorant belief state by a series of expansions. In PPT, constraining, but not conditioning, has the analogous property. This is one of the main reasons we prefer to constraining and not conditioning to be the probabilistic version of expansion. (Voorbraak, 1999, p. 4)

But Voorbraak does not address the issue that C and ϕ are different kinds of observations, so constraining, as defined here, cannot be an alternative to conditioning. C cannot be used directly for constraining and ϕ cannot be used directly for conditioning.

W.l.o.g., we can assume C is represented by α. If we take $b \,\mathsf{GI}\, \alpha$ to be an expansion operation whenever $b(\alpha) > 0$, then one might ask, Is it possible to obtain any belief base B' from the ignorant belief base $B = \emptyset$ by a series of expansions, using our approach? The answer is, No. For instance, there is no observation or series of observations which can change $B = \{\}$ into $B' = \{(q) \geq 0.6\}$. But if we *were* to allow sentences (constraints) in L^{prob} to be observations, then we *could* obtain any B' from the ignorant B.

Grove and Halpern (1998) investigate what "update" (incorporation of an observation with current beliefs, such that the observation does not contradict the beliefs) means in a framework where beliefs are represented by a set of belief states. They state that the main purpose of their paper is to illustrate how different the set-of-distributions framework can be, "technically", from the standard single-distribution framework. They propose six postulates characterizing what properties an update function should have. They say that some of the postulates are obvious, some arguable and one probably too strong. Out of seven (families of) update functions only the one based on conditioning ($Upd_{cond}(\cdot)$) and the one based on constraining ($Upd_{constrain}(\cdot)$) satisfy all six postulates, where $Upd_{cond}(\Pi^{B}, \alpha) := \{(b \,\mathsf{BC}\, \alpha) \in \Pi \mid b \in \Pi^{B}, b(\alpha) > 0\}$ and where they interpret Voorbraak's (1999) constraining as $Upd_{constrain}(\Pi^{B}, \alpha) := \{b \in \Pi^{B} \mid b(\alpha) = 1\}$. Grove and Halpern (1998) do not investigate the case when an observation must be incorporated while it is (possibly) inconsistent with the old beliefs (i.e., revision).

Kern-Isberner (2001) develops a new perspective of probabilistic belief change. Based on the ideas of Alchourrón, Gärdenfors, and Makinson (1985) and Katsuno and Mendelzon (1991) (KM), the operations of revision and update, respectively, are investigated within a probabilistic framework. She employs as basic knowledge structure a belief base $(b, \mathcal{R})$, where b is a probability distribution (belief state) of background knowledge and $\mathcal{R}$ is a set of probabilistic conditionals of the form $A \rightsquigarrow B[x]$ meaning 'The probability of B, given A, is x. A universal inference operation – based on the techniques of optimum entropy – is introduced as an "adequate and powerful method to realize probabilistic belief change".

By having a belief state available in the belief base, minimum cross-entropy can be used. The intention is then that an agent with belief base $(b, \mathcal{T})$ should always reason w.r.t. belief state $b^{\mathcal{T}} := \arg\min_{c \in \Pi, c \Vdash \mathcal{T}} R(c, b)$. Kern-Isberner (2001) then defines the probabilistic belief revision of $(b, \mathcal{R})$ by evidence $\mathcal{S}$ as $(b, \mathcal{R} \cup \mathcal{S})$. And the probabilistic belief update of $(b, \mathcal{R})$ by evidence $\mathcal{S}$ is defined as $(b^{\mathcal{R}}, \mathcal{S})$.[5] She distinguishes between revision as a knowledge adding process, and updating as a change-recording process. Kern-Isberner (2001) sets up comparisons of maximum cross-entropy belief change with AGM revision and KM update. Cases where, for update, new information $\mathcal{R}$ is inconsistent with the prior distribution b, or, for revision, is inconsistent with b or the context $\mathcal{R}$, are not dealt with (Kern-Isberner, 2001, p. 399, 400).

Having a belief state available for modification when new evidence is to be adopted is quite convenient. As Voorbraak (1999) argues, however, an agent's ignorance can hardly be represented in an epistemic state where a single belief state must always be chosen.

The reader may also refer to a later paper (Kern-Isberner, 2008) in which many of the results of the work just reviewed are generalized to belief bases of the form $(\Psi, \mathcal{R})$, where Ψ denotes a general *epistemic state*. In that paper, she considers two instantiations of Ψ, namely as a *probability distribution* and as an *ordinal conditional function* (first introduced by Spohn (1988)).

Yue and Liu (2008) propose a probabilistic revision operation for imprecise probabilistic beliefs in the framework of Probabilistic Logic Programming (PLP). New evidence may be a probabilistic (conditional) formula and needs not be consistent with the original beliefs. Revision via imaging (e.g., GI) also overcomes this consistency issue.

Essentially, their *probabilistic epistemic states* Ψ are induced from a PLP program which is a set of formulae, each formula having the form $(\psi \mid \phi)[l, u]$, meaning that the probability of the conditional $(\psi \mid \phi)$ lies in the interval $[l, u]$.

The operator they propose has the characteristic that if an epistemic state Ψ represents a single probability distribution, revising collapses to Jeffrey's rule and Bayesian conditioning.

They mention that it is required that the models (distributions) of Ψ is a convex set. There might thus be an opportunity to employ their revision operation on a representative set of boundary distributions as proposed in this paper.

[5] This is a very simplified version of what she presents. Please refer to the paper for details.

Conclusion

In this paper, we propose an approach how to generate a new probabilistic belief base from an old one, given a new piece of non-probabilistic information, where a belief base is a finite set of sentences, each sentence stating the likelihood of a proposition about the world. In this framework, an agent's belief base represents the set of belief states compatible with the sentences in it. In this sense, the agent is able to represent its knowledge *and* ignorance about the true state of the world.

We used a version of the so-called *imaging* approach to implement the revision operation.

Two methods were proposed: revising a finite set of 'boundary belief states' and revising a least biased belief state. We focussed on the former and showed that the latter gives different results.

There were two main contribution of this paper. The first was to prove that the set of belief states satisfying B_{new} is exactly those belief states satisfying the original belief base, revised. The second was to uncover an interesting conflict in the results of the two belief base revision methods. It is worth further understanding the reasons behind such a difference, as such an investigation could give more insight about the mechanisms behind the two methods and indicate possible pros and cons of each.

Acknowledgements

The work of Giovanni Casini has been supported by the Fonds National de la Recherche, Luxembourg, and cofunded by the Marie Curie Actions of the European Commission (FP7-COFUND) (AFR/9181001).

References

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50(2):510–530.

Boutilier, C. 1998. A unified model of qualitative belief change: a dynamical systems perspective. *Artificial Intelligence* 98(1–2):281–316.

Chhogyal, K.; Nayak, A.; Schwitter, R.; and Sattar, A. 2014. Proceedings of the thirteenth pacific rim international conference on artificial intelligence (pricai 2014). In Pham, D., and Park, S., eds., *Proc. of PRICAI 2014*, volume 8862 of *LNCS*, 694–707. Springer-Verlag.

Cover, T., and Thomas, J. 1991. *Elements of Information Theory*. New York: Wiley.

Csiszár, I. 1975. I-divergence geometry of probability distributions and minimization problems. *Annals of Probability* 3:146–158.

Dubois, D., and Prade, H. 1993. Belief revision and updates in numerical formalisms: An overview, with new results for the possibilistic framework. In *Proceedings of the 13th International Joint Conference on Artifical Intelligence*, volume 1 of *IJCAI'93*, 620–625. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Gärdenfors, P. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Massachusetts/England: MIT Press.

Goldszmidt, M.; Morris, P.; and Pearl, J. 1990. A maximum entropy approach to nonmonotonic reasoning. In *Proceedings of the Eighth Natl. Conf. on Artificial Intelligence (AAAI-90)*, 646–652. AAAI Press.

Grove, A., and Halpern, J. 1998. Updating sets of probabilities. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, 173–182. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Hunter, D. 1991. Maximum entropy updating and conditionalization. In Spohn, W.; Van Fraassen, B.; and Skyrms, B., eds., *Existence and Explanation*, volume 49 of *The University of Western Ontario Series in Philosophy of Science*. Springer Netherlands. 45–57.

Jaynes, E. 1978. Where do we stand on maximum entropy? In *The Maximum Entropy Formalism*. MIT Press. 15–118.

Katsuno, H., and Mendelzon, A. 1991. On the difference between updating a knowledge base and revising it. In *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, 387–394.

Kern-Isberner, G., and Rdder, W. 2004. Belief revision and information fusion on optimum entropy. *International Journal of Intelligent Systems* 19(9):837–857.

Kern-Isberner, G. 1998. Characterizing the principle of minimum cross-entropy within a conditional-logical framework. *Artificial Intelligence* 98(12):169 – 208.

Kern-Isberner, G. 2001. Revising and updating probabilistic beliefs. In Williams, M.-A., and Rott, H., eds., *Frontiers in Belief Revision*, volume 22 of *Applied Logic Series*. Kluwer Academic Publishers, Springer Netherlands. 393–408.

Kern-Isberner, G. 2008. Linking iterated belief change operations to nonmonotonic reasoning. In *Proceedings of the Eleventh International Conference on Principles of Knowledge Representation and Reasoning*, 166–176. Menlo Park, CA: AAAI Press.

Kullback, S. 1968. *Information theory and statistics*, volume 1. New York: Dover, 2nd edition.

Lehmann, D.; Magidor, M.; and Schlechta, K. 2001. Distance semantics for belief revision. *Journal of Symboloc Logic* 66(1):295–317.

Lewis, D. 1976. Probabilities of conditionals and conditional probabilities. *Philosophical Review* 85(3):297–315.

Paris, J., and Vencovsk, A. 1997. In defense of the maximum entropy inference process. *International Journal of Approximate Reasoning* 17(1):77–103.

Paris, J. 1994. *The Uncertain Reasoner's Companion: A Mathematical Perspective*. Cambridge: Cambridge University Press.

Poole, D., and Mackworth, A. 2010. *Artificial Intelligence: Foundations of Computational Agents*. New York, USA: Cambridge University Press.

Rens, G., and Meyer, T. 2015. A new approach to probabilistic belief change. In Russell, I., and Eberle, W., eds., *Proceedings of the International Florida AI Research Society Conference (FLAIRS)*, 582–587. AAAI Press.

Shore, J., and Johnson, R. 1980. Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *Information Theory, IEEE Transactions on* 26(1):26–37.

Spohn, W. 1988. Ordinal conditional functions: A dynamic theory of epistemic states. In Harper, W., and Skyrms, B., eds., *Causation in Decision, Belief Change, and Statistics*, volume 42 of *The University of Western Ontario Series in Philosophy of Science*. Springer Netherlands. 105–134.

Voorbraak, F. 1999. Partial Probability: Theory and Applications. In *Proceedings of the First International Symposium on Imprecise Probabilities and Their Applications*, 360–368. url: decsai.ugr.es/ smc/isipta99/proc/073.html.

Yue, A., and Liu, W. 2008. Revising imprecise probabilistic beliefs in the framework of probabilistic logic programming. In *Proceedings of the Twenty-third AAAI Conf. on Artificial Intelligence (AAAI-08)*, 590–596.

Reactive Policies with Planning for Action Languages [*]

Zeynep G. Saribatur and **Thomas Eiter**
Institut für Informationssysteme, Technische Universität Wien
Favoritenstraße 9-11, A-1040 Vienna, Austria
{zeynep,eiter}@kr.tuwien.ac.at

Abstract

We describe a representation in a high-level transition system for policies that express a reactive behavior for the agent. We consider a target decision component that figures out what to do next and an (online) planning capability to compute the plans needed to reach these targets. Our representation allows one to analyze the flow of executing the given reactive policy, and to determine whether it works as expected. Additionally, the flexibility of the representation opens a range of possibilities for designing behaviors.

Autonomous agents are systems that decide for themselves what to do to satisfy their design objectives. These agents have a knowledge base that describes their capabilities, represents facts about the world and helps them in reasoning about their course of actions. A reactive agent interacts with its environment. It perceives the current state of the world through sensors, consults its memory (if there is any), reasons about actions to take and executes them in the environment. A policy for these agents gives guidelines to follow during their interaction with the environment.

As autonomous systems become more common in our lives, the issue of verifying that they behave as intended becomes more important. During the operation of an agent, one would want to be sure that by following the designed policy, the agent will achieve the desired results. It would be highly costly, time consuming and sometimes even fatal to realize at runtime that the designed policy of the agent does not provide the expected properties.

For example, in search and rescue scenarios, an agent needs to find a missing person in unknown environments. A naive approach would be to directly try to find a plan that achieves the main goal of finding the person. However, this problem easily becomes troublesome, since not knowing the environment causes the planner to consider all possible cases and find a plan that guarantees reaching the goal in all settings. Alternatively, one can describe a reactive policy for the agent that determines its course of actions according to its current knowledge, and guides the agent in the environment towards the main goal. A possible such policy could be "always move to the farthest unvisited point in visible distance, until a person is found". Following this reactive policy, the agent would traverse the environment by choosing its actions to reach the farthest possible point from the current state, and by reiterating the decision process after reaching a new state. The agent may also remember the locations it has been in and gain information (e.g. obstacle locations) through its sensors on the way. Verifying beforehand whether or not the designed policy of the agent satisfies the desired goal (e.g. can the agent always find the person?), in all possible instances of the environment is nontrivial.

Action languages (Gelfond and Lifschitz 1998) provide a useful framework on defining actions and reasoning about them, by modeling dynamic systems as transition systems. Their declarative property helps in describing the system in an understandable, concise language, and they also address the problems encountered when reasoning about actions. By design, these languages are made to be decidable, which ensures reliable descriptions of dynamic systems. As these languages are closely related with classical logic and answer set programming (ASP) (Lifschitz 2008; 1999), they can be translated into logic programs and queried for computation. The programs produced by such translations can yield sound and complete answers to such queries. There have been various works on action languages (Gelfond and Lifschitz 1998; 1993; Giunchiglia and Lifschitz 1998) and their reasoning systems (Giunchiglia et al. 2004; Gebser, Grote, and Schaub 2010), with underlying mechanisms that rely on SAT and ASP solvers.

The shortage of representations that are capable of modeling reactive policies prevents one from verifying such policies using action languages as above before putting them into use. The necessity of such a verification capability motivates us to address this issue. We thus aim for a general model that allows for verifying the reactive behavior of agents in environments with different types in terms of observability and determinism. In that model, we want to use the representation power of the transition systems described by action languages and combine components that are efficient for describing reactivity.

Towards this aim, we consider in this paper agents with a reactive behavior that decide their course of actions by determining targets to achieve during their interaction with the environment. Such agents come with an (online) planning

[*] This work has been supported by Austrian Science Fund (FWF) project W1255-N23.

capability that computes plans to reach the targets. This method matches the observe-think-act cycle of Kowalski and Sadri (1999), but involves a planner that considers targets. The flexibility in the two components - target development and external planning - allow for a range of possibilities for designing behaviors. For example, one can use HEX (Eiter et al. 2005) to describe a program that determines a target given the current state of an agent, finds the respective plan and the execution schedule. ACTHEX programs (Fink et al. 2013), in particular, provide the tools to define such reactive behaviors as it allows for iterative evaluation of the logic programs and the ability to observe the outcomes of executing the actions in the environment. Specifically, we make the following contributions:

(1) We introduce a novel framework for describing the semantics of a policy that follows a reactive behavior, by integrating components of target establishment and online planning. The purpose of this work is not synthesis, but to lay foundations for verification of behaviors of (human-designed) reactive policies. The outsourced planning might also lend itself for modular, hierarchic planning, where macro actions (expressed as targets) are turned into a plan of micro actions. Furthermore, outsourced planning may also be exploited to abstract from correct sub-behaviors (e.g. going always to the farthest point).

(2) We relate this to action languages and discuss possibilities for policy formulation. In particular, we consider the action language $\mathcal{C}$ (Giunchiglia and Lifschitz 1998) to illustrate an application.

The remainder of this paper is organized as follows. After some preliminaries, we present a running example and then the general framework for modeling policies with planning. After that, we consider the relation to action languages, and as a particular application we consider (a fragment of) the action language $\mathcal{C}$. We briefly discuss some related work and conclude with some issues for ongoing and future work.

Preliminaries

Definition 1. A *transition system* $\mathcal{T}$ is defined as $\mathcal{T} = \langle S, S_0, \mathcal{A}, \Phi \rangle$ where

- S is the set of states.
- $S_0 \subseteq S$ is the set of possible initial states.
- $\mathcal{A}$ is the set of possible actions.
- $\Phi : S \times \mathcal{A} \to 2^S$ is the transition function, returns the set of possible successor states after applying a possible action in the current state.

For any states $s, s' \in S$, we say that there is a *trajectory* between s and s', denoted by $s \to^{\sigma} s'$ for some action sequence $\sigma = a_1, \ldots, a_n$ where $n \geq 0$, if there exist $s_0, \ldots, s_n \in S$ such that $s = s_0, s' = s_n$ and $s_{i+1} \in \Phi(s_i, a_{i+1})$ for all $0 \leq i < n$.

We will refer to this transition system as the *original* transition system. The constituents S and $\mathcal{A}$ are assumed to be finite in the rest of the paper. Note that, this transition system represents fully observable settings. Large environments cause high number of possibilities for states, which cause the transition systems to be large. Especially, if the environment is nondeterministic, the resulting transition system contains high amount of transitions between states, since one needs to consider all possible outcomes of executing an action.

Action Languages

Action languages describe a particular type of transition systems that are based on action signatures. An *action signature* consists of a set $\mathbf{V}$ of value names, a set $\mathbf{F}$ of fluent names and a set $\mathbf{A}$ of action names. Any *fluent* has a *value* in any *state of the world*.

A transition system of an action signature $\langle \mathbf{V}, \mathbf{F}, \mathbf{A} \rangle$ is similar to Defn. 1, where $\mathcal{A} = \mathbf{A}$ and Φ corresponds to the relation $R \subseteq S \times \mathbf{A} \times S$. In addition, we have a value function $V : \mathbf{F} \times S \to \mathbf{V}$, where $V(P, s)$ shows the *value of* P in state s. A transition system can be thought as a labeled directed graph, where a state s is represented by a vertex labeled with the function $P \to V(P, s)$, that gives the value of the fluents. Every triple $\langle s, a, s' \rangle \in R$ is represented by an edge leading from a state s to a state s' and labeled by a.

An action a is *executable* at a state s, if there is at least one state s' such that $\langle s, a, s' \rangle \in R$ and a is *deterministic* if there is at most one such state. Concurrent execution of actions can be defined by considering transitions in the form $\langle s, A, s' \rangle$ with a set $A \subseteq \mathbf{A}$ of actions, where each action $a \in A$ is executable at s.

An action signature $\langle \mathbf{V}, \mathbf{F}, \mathbf{A} \rangle$ is *propositional* if its value names are truth values: $\mathbf{V} = \{\mathsf{f}, \mathsf{t}\}$. In this work, we confine to propositional action signatures.

The transition system allows one to answer queries about the program. For example, one can find a plan to reach a goal state from an initial state, by searching for a path between the vertices that represent these states in the transition system. One can express properties about the paths of the transition system by using an action query language.

Running Example: Search Scenarios

Consider a memoryless agent that can sense horizontally and vertically, in an unknown $n \times n$ grid cell environment with obstacles, where a missing person needs to be found. Suppose we are given the action description of the agent with a policy of "always going to the farthest reachable point in visible distance (until a person is found)". Following this reactive policy, the agent chooses its course of actions to reach the farthest reachable point, referred as *target*, from its current location with respect to its current knowledge about the environment. After executing the plan and reaching a state that satisfies the target, the decision process is reiterated and a new target, hence a new course of actions, is determined.

Given such a policy, one would want to check whether or not the agent can always find the person, in all instances of the environment. Note that we assume that the obstacles are placed in a way that the person is always reachable.

Figure 1 shows some possible instances for n=3, where the square in a cell represents the agent and the dot represents the missing person. The course of actions determined by the policy in all the instances is to move to (3,1), which is the

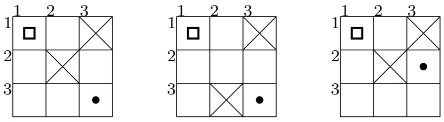


Figure 1: Possible instances of a search scenario

farthest reachable point, i.e. target. It can be seen that (a) is an instance where the person can be found with the given policy, while in (b) the agent goes in a loop and can't find the person, since after reaching (3,1) it will decide to move to (1,1) again. In (c), after reaching (3,1) following its policy, the agent has two possible directions to choose, since there are two farthest points. It can either move to (3,3), which would result in seeing the person, or it can move back to (1,3), which would mean that there is a possibility for the agent to go in a loop.

Notice that our aim is different from *finding a policy* (i.e. global plan) that satisfies certain properties (i.e. goals) in an unknown environment. On the contrary, we assume that we are given a representation of a system with a certain policy, and we want to check what it is capable (or incapable) of.

Modeling Policies in Transition Systems

A reactive policy is described to reach some main goal, by guiding the agent through its interaction with the environment. This guidance can be done by determining the course of actions to bring about targets from the current situation, via externally computed plans. A transition system that models such policies should represent the flow of executing the policy, which is the agent's actual trajectory in the environment following the policy. This would allow for verifying whether execution of a policy results in reaching the desired main goal, i.e. the policy works.

We define such a transition system by clustering the states into groups depending on a *profile*. A profile is determined by evaluating a set of formulas over a state that informally yield attribute (respectively feature) values; states with the same attribute values are clustered into one. The choice of formulas for determining profiles depends on the given policy or the environment one is considering. Then, the transitions between these clusters are defined according to the policy. The newly defined transitions are able to show the evaluation of the policy by a higher level action from one state to the next state. This next state satisfies the target determined by a *target component*, and the higher level action corresponds to the execution of an externally computed plan.

Having such a classification on states and defining higher level transitions between the states can help in reducing the state space or the number of transitions when compared to the original transition system. Furthermore, it aids in abstraction and allows one to emulate a modular hierarchic approach, in which a higher level (macro) action, expressed by a target, is realized in terms of a sequence of (micro) actions that is compiled by the external planner, which may use different ways (planning on the fly, resorting to scripts etc.)

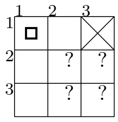


Figure 2: A cluster of states

State profiles according to the policy

We now describe a classification of states, which helps to omit parts of the state that are irrelevant with respect to the environment or the policy. This classification is done by determining profiles, and clustering the states accordingly.

Example 1. Remember the possible instances from Figure 1 in the running example. Due to partial observability, the agent is unable to distinguish the states that it is in, and the unobservable parts are irrelevant to the policy. Now assume that there are fluents that hold the information of the agent's location, the locations of the obstacles and the reachable points. One can determine a profile of the form "the agent is at (1,1), sees an obstacle at (1,3), and is able to reach to points at (1,2), (2,1), (3,1)" by not considering the remaining part of the environment that the agent can not observe. The states that have this profile can be clustered in one group as in Figure 2, where the cells with question marks demonstrate that they are not observable by the agent.

For partially observable environments, the notion of indistinguishable states can be used in the classification of states. The states that provide the same observations for the agent are considered as having the same profile. However, in fully observable environments, observability won't help in reducing the state space. One needs to find other notions to determine profiles.

We consider a *classification function*, $h : S \to \Omega_h$, where Ω_h is the set of possible state clusters. This is a general notion applicable to fully and partially observable cases.

Definition 2. An *equalized state* relative to the classification function h is a state $\hat{s} \in \Omega_h$. The term *equalized* comes from the fact that the states in the same classification are considered as the same, i.e. equal.

To talk about a state s that is clustered into an equalized state $\hat{s}$, we use the notation $s \in \hat{s}$, where we identify $\hat{s}$ with its pre-image (i.e. the set of states that are mapped to $\hat{s}$ according to h).

Different from the work by Son and Baral (2001) where they consider a "combined-state" which consists of the real state of the world and the states that the agent thinks it may be in, we consider a version where we combine the real states into one state if they provide the same classification (or observation, in case of partial observability) for the agent. The equalization of states allows for omitting the details that are irrelevant to the behavior of the agent.

Transition systems according to the policy

We now define the notion of a transition system that is able to represent the evaluation of the policy on the state clusters.

Definition 3. An *equalized (higher level) transition system* $\mathcal{T}_h$, with respect to the classification function h, is defined as $\mathcal{T}_h = \langle \widehat{S}, \widehat{S}_0, G_B, \mathcal{B}, \Phi_{\mathcal{B}} \rangle$, where

- $\widehat{S}$ is the finite set of equalized states;
- $\widehat{S}_0 \subseteq \widehat{S}$ is the finite set of initial equalized states, where $\hat{s} \in \widehat{S}_0$ if there is some $s_i \in \hat{s}$ such that $s_i \in S_0$ holds;
- G_B is the finite set of possible *targets* relative to the behavior, where a target can be satisfied by more than one equalized state;
- $\mathcal{B} : \widehat{S} \to 2^{G_B}$, is the *target function* that returns the possible targets to achieve from the current equalized state, according to the policy;
- $\Phi_{\mathcal{B}} : \widehat{S} \to 2^{\widehat{S}}$ is the transition function according to the policy, referred to as the *policy execution function*, returns the possible resulting equalized states after applying the policy in the current equalized state.

The target function gets the equalized state as input and produces the possible targets to achieve. These targets may be expressed as formulas over the states (in particular, of states that are represented by fluents or state variables), or in some other representation. A target can be considered as a subgoal condition to hold at the follow-up state, depending on the current equalized state. The aim would be to intend to reach a state that satisfies the conditions of the target, without paying attention to the steps taken in between. That's where the policy execution function comes into the picture.

The formal description of the policy execution function is as follows:

$$\begin{aligned}\Phi_B(\hat{s}) = \{\hat{s}' \mid {} & \hat{s}' \in \mathit{Res}(\hat{s}, \sigma), \\ & \sigma \in \mathit{Reach}(\hat{s}, g_B), g_B \in \mathcal{B}(\hat{s})\},\end{aligned}$$

where Reach is an outsourced function that returns a plan $\sigma = \langle a_1, \ldots, a_n \rangle, n \geq 0$ needed to reach a state that meets the conditions g_B from the current equalized state $\hat{s}$:

$$\mathit{Reach}(\hat{s}, g_B) \subseteq \{\sigma \mid \forall \hat{s}' \in \mathit{Res}(\hat{s}, \sigma) : \hat{s}' \models g_B\}$$

where $\hat{s} \models g_B \Leftrightarrow \forall s \in \hat{s} : s \models g_B$, and Res gives the resulting states of executing a sequence of actions at a state $\hat{s}$:

$$\begin{aligned}&\mathit{Res}(\hat{s}, \langle a_1, \ldots, a_{n \geq 1} \rangle) \quad = \\ &\quad \begin{cases} \bigcup_{\hat{s}' \in \hat{\Phi}(\hat{s}, a_1)} \mathit{Res}(\hat{s}', \langle a_2, \ldots, a_n \rangle) & \hat{\Phi}(\hat{s}, a_1) \neq \emptyset \\ \{\hat{s}_{err}\} & \hat{\Phi}(\hat{s}, a_1) = \emptyset \end{cases} \\ &\mathit{Res}(\hat{s}, \langle\rangle) \qquad\qquad\quad = \{\hat{s}\}\end{aligned}$$

where the state $\hat{s}_{err}$ is an artifact state that does not satisfy any of the targets, and

$$\hat{\Phi}(\hat{s}, a) = \{\hat{s}' \mid \exists s' \in \hat{s}' \; \exists s \in \hat{s} : s' \in \Phi(s, a)\}.$$

Figure 3 demonstrates a transition in the equalized transition system. The equalized states may contain more than one state that has the same profile. Depending on the current state, $\hat{s}$, the policy chooses the next target, g_B, that should be satisfied. There may be more than one equalized state satisfying the same target. The policy execution function $\Phi_B(\hat{s})$ finds a transition into one of these equalized states, $\hat{s}'$, that is reachable from the current equalized state. The transition Φ_B is considered as a big jump between states, where the actions taken and the states passed in between are omitted.

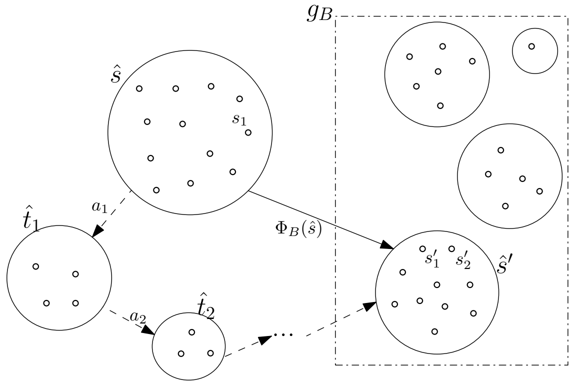


Figure 3: A transition in the equalized transition system

Notice that we assume that the outsourced Reach function is able to return conformant plans that guarantee to reach a state that satisfies the determined targets. In particular, σ may also contain only one action. For practical reasons, we consider Reach to be able to return a subset of all conformant plans. The maximal possible Reach, where we have equality, is denoted with Reach_0.

Consider the case of uncertainty, where the agent requires to do some action, e.g. $\mathit{checkDoor}$, in order to get further information about its state. One can define the target function to return as target a dummy fluent to ensure that the action is made, e.g. $\mathit{doorIsChecked}$, and given this target, the Reach function can return the desired action as the plan. The nondeterminism or partial observability of the environment is modeled through the set of possible successor states returned by Res.

The generic definition of the equalized transition system allows for the possibility of representing well-known concepts like purely reactive systems or universal planning (Cimatti, Riveri, and Traverso 1998a). To represent reactive systems, one can describe a policy of "pick some action". This way one can model reactive systems that do not do reasoning, but immediately react to the environment with an action. As for the exactly opposite case, which is finding a plan that guarantees reaching the goal, one can choose the target as the main goal. Then, the Reach would have the difficult task of finding a universal plan or a conformant plan that reaches the main goal. If however, one is aware of such a plan, then it is possible to mimic the plan by modifying the targets G_B and the target function $\mathcal{B}$ in a way that at each point in time the next action in the plan is returned by Reach, and the corresponding transition is made. For that, one needs to record information in the states and keep track of the targets.

As the function Reach is outsourced, we rely on an implementation that returns conformant plans for achieving transitions in the equalized transition systems. This naturally raises the issue of whether a given such implementation is suitable, and leads to the question of soundness (only correct plans are output) and completeness (some plan will be output, if one exists). We next assess how expensive it is to test this, under some assumptions about the representation and computational properties of (equalized) transition systems, which will then also be used for assessing the cost of policy checking.

Assumptions We have certain assumptions on the representation of the system. We assume that given a state $s \in S$ which is implicitly given using a binary encoding, the cost of evaluating the classification $h(s)$, the (original) transition $\Phi(s,a)$ for some action a, and recognizing the initial state, say with $\Phi_{init}(s)$, is polynomial. The cost could also be in NP, if projective (i.e. existentially quantified) variables are allowed. Furthermore, we assume that the size of the representation of a "target" in G_B is polynomial in size of the state, so that given a string, one can check in polynomial time if it is a correct target description g_B. This test can also be relaxed to be in NP by allowing projective variables.

Given these assumptions, we have the following two results. These results show the cost of checking whether an implementation of *Reach* that we have at hand is sound (delivers correct plans) and in case does not skip plans (is complete); we assume here that testing whether $\sigma \in \mathit{Reach}(\hat{s}, g_B)$ is feasible in Π_2^p (this is the cost of verifying conformant plans, and we may assume that *Reach* is no worse than a naive guess and check algorithm).

Theorem 1 (soundness of *Reach*)**.** *Let* $\mathcal{T}_h = \langle \widehat{S}, \widehat{S}_0, G_B, \mathcal{B}, \Phi_\mathcal{B}\rangle$ *be a transition system with respect to a classification function h. The problem of checking whether every transition found by the policy execution function* $\Phi_\mathcal{B}$ *induced by a given implementation Reach is correct is in* Π_3^p.

Proof (Sketch). According to the definition of the policy execution function, every transition from a state $\hat{s}$ to some state $\hat{s}'$ corresponds to some plan σ returned by $\mathit{Reach}(\hat{s}, g_B)$. Therefore, first one needs to check whether each plan $\sigma = \langle a_1, a_2, \ldots, a_n\rangle$ returned by *Reach* given some $\hat{s}$ and g_B is correct. For that we need to check two conditions on the corresponding trajectories of the plan:

(i) for all partial trajectories $\hat{s}_0, \hat{s}_1, \ldots, \hat{s}_{i-1}$ it holds that for the upcoming action a_i from the plan σ, $\hat{\Phi}(\hat{s}_{i-1}, a_i) \neq \emptyset$ (i.e. the action is applicable)

(ii) for all trajectories $\hat{s}_0, \hat{s}_1, \ldots, \hat{s}_n$, $\hat{s}_n \models g_B$.

Checking whether these conditions hold is in Π_2^p.

Thus, to decide whether for some state $\hat{s}$ and target g_B the function $\Phi_B(\hat{s}, g_B)$ does not work correctly, we can guess $\hat{s}$ (resp. $s \in \hat{s}$), g_B and a plan σ and verify that $\sigma \in \mathit{Reach}(\hat{s}, g_B)$ and that σ is not correct. As the verification is doable with an oracle for Σ_2^p in polynomial time, a counterexample for correctness can be found in Σ_3^p; thus the problem is in Π_3^p. □

The complexity is lower, if output checking of *Reach* has lower complexity (in particular, it drops to Π_2^p if output checking is in co-NP).

The result for soundness of *Reach* is complemented with another result for completeness with respect to short (polynomial size) conformant plans that are returned by *Reach*.

Theorem 2 (completeness of *Reach*)**.** *Let* $\mathcal{T}_h = \langle \widehat{S}, \widehat{S}_0, G_B, \mathcal{B}, \Phi_\mathcal{B}\rangle$ *be a transition system with respect to a classification function h. Deciding whether for a given implementation Reach,* Φ_B *fulfills* $\hat{s}' \in \Phi_B(\hat{s})$ *whenever a short conformant plan from* $\hat{s}$ *to* $\hat{s}'$ *exists in* T_h*, is in* Π_4^p.

Proof (Sketch). For a counterexample, we can guess some $\hat{s}$ and $\hat{s}'$ (resp. $s \in \hat{s}$, $s' \in \hat{s}'$) and some short plan σ and verify that (i) σ is a valid conformant plan in $\mathcal{T}_h$ to reach $\hat{s}'$ from $\hat{s}$, and (ii) that a target g_B exists such that $\mathit{Reach}(\hat{s}, g_B)$ produces some output. We can verify (i) using a Π_2^p oracle to check that σ is a conformant plan, and we can verify (ii) using a Π_3^p oracle (for all guesses of targets g_B and short plans σ', either g_B is not a target for $\hat{s}$ or σ' is not produced by $\mathit{Reach}(\hat{s}, g_B)$). This establishes membership in Π_4^p. □

As in the case of soundness, the complexity drops if checking the output of *Reach* is lower (in particular, to Π_3^p if the output checking is in co-NP).

We also restrict the plans σ that are returned by $\mathit{Reach}(\hat{s}, g_B)$ to have polynomial size. This constraint would not allow for exponentially long conformant plans (even if they exist). Thus, the agent is forced under this restriction to develop targets that it can reach in polynomially many steps, and then to go on from these targets. Informally, this does not limit the capability of the agent in general. The "long" conformant plans can be split into short plans with a modified policy and by encoding specific targets into the states.

We denote the main goal that the reactive policy is aiming for by g_∞. Our aim is to have the capability to check whether following the policy always results in reaching some state that satisfies the main goal. That is, for each run, i.e. sequence $\hat{s}_0, \hat{s}_1, \ldots$ such that $\hat{s}_0 \in \widehat{S}_0$ and $\hat{s}_{i+1} \in \Phi_B(\hat{s}_i)$, for all $i \geq 0$, there is some $j \geq 0$ such that $\hat{s}_j \models g_\infty$. (The behavior could be easily modified to stop or to loop in any state $\hat{s}$ that satisfies the goal.) This way we can say whether the policy works or not. Under the assumptions from above, we obtain the following

Theorem 3. *The problem of determining that the policy works is in PSPACE.*

Proof (Sketch). One needs to look at all runs $\hat{s}_0, \hat{s}_1, \ldots$ from every initial state $\hat{s}_0$ in the equalized transition system and check whether each such run has some state $\hat{s}_j$ that satisfies the main goal g_∞. Given that states have a representation in terms of fluent or state variables, there are at most exponentially many different states. Thus to find a counterexample, a run of at most exponential length in which g_∞ is not satisfied is sufficient. Such a run can be nondeterministically built in polynomial space; as NPSPACE = PSPACE, the result follows. □

Note that in this formulation, we have tacitly assumed that the main goal can be established in the original system, thus at least some trajectory from some initial state to a state fulfilling the goal exists (this can be checked in PSPACE as well). In a more refined version, we could define the working of a policy relative to the fact that some abstract plan would exist that makes g_∞ true; naturally, thus may impact the complexity of the policy checking.

Above, we have been considering arbitrary states, targets and transitions in the equalized transition system. In fact, for the particular behavior, only the states that can be encountered in runs really matter; these are the reachable states defined as follows.

Definition 4. A state $\hat{s}$ is *reachable* from an initial state in the equalized transition system if and only if $s \in \mathcal{R}_i$ for some $i \in \mathbb{N}$ where $\mathcal{R}_i$ is defined as follows.

$$\begin{array}{ll} \mathcal{R}_0 & = \widehat{S}_0 \\ \mathcal{R}_{i+1} & = \bigcup_{\hat{s}\in\mathcal{R}_i} \Phi_B(\hat{s}) \\ & \dots \\ \mathcal{R}^\infty & = \bigcup_{i\geq 0} \mathcal{R}_i. \end{array}$$

Under the assumptions that apply to the previous results, we can state the following.

Theorem 4. *The problem of determining whether a state in an equalized transition system is reachable is in PSPACE.*

The notions of soundness and completeness of an outsourced planning function *Reach* could be restricted to reachable states; however, this, would not change the cost of testing these properties in general (assuming that $\hat{s} \in \mathcal{R}$ is decidable with sufficiently low complexity).

Constraining equalization

The definition of $\hat{\Phi}$ allows for certain transitions between equalized states that don't have corresponding concrete transitions in the original transition system. However, the aim of defining such an equalized transition system is not to introduce new features, but to keep the structure of the original transition system and discard the unnecessary parts with respect to the policy. Therefore, one needs to give further restrictions on the transitions of the equalized transition system, in order to obtain the main objective.

Let us consider the following condition

$$\hat{s}' \in \hat{\Phi}(\hat{s}, a) \Leftrightarrow \forall s' \in \hat{s}',\ \exists s \in \hat{s} :\ s' \in \Phi(s, a) \qquad (1)$$

This condition ensures that a transition between two states $\hat{s}_1, \hat{s}_2$ in the equalized transition system represents that any state in $\hat{s}_2$ has a transition from some state in $\hat{s}_1$. An equalization is called *proper* if condition (1) is satisfied.

Theorem 5. *Let $\mathcal{T}_h{=}\langle \widehat{S}, \widehat{S}_0, G_B, \mathcal{B}, \Phi_\mathcal{B}\rangle$ be a transition system with respect to a classification function h. Let $\hat{\Phi}$ be the transition function that the policy execution function $\Phi_\mathcal{B}$ is based on. The problem of checking whether $\hat{\Phi}$ is proper is in Π^p_2.*

Proof (sketch). As a counterexample, one needs to guess $\hat{s}, a,\ \hat{s}' \in \hat{\Phi}(\hat{s}, a)$ and $s' \in \hat{s}'$ such that no $s \in \hat{s}$ has $s' \in \Phi(s, a)$. □

The results in Theorems 1-5 are all complemented by lower bounds for realistic realizations of the parameters (notably, for typical action languages such as fragments of $\mathcal{C}$).

The following proposition is based on the assumption that the transition function $\hat{\Phi}$ satisfies condition (1).

Proposition 1 (soundness). *Let $\mathcal{T}_h{=}\langle \widehat{S}, \widehat{S}_0, G_B, \mathcal{B}, \Phi_\mathcal{B}\rangle$ be a transition system with respect to a classification function h. Let $\hat{s}_1, \hat{s}_2 \in \widehat{S}$ be equalized states that are reachable from some initial states, and $\hat{s}_2 \in \Phi_B(\hat{s}_1)$. Then for any concrete state $s_2 \in \hat{s}_2$ there is a concrete state $s_1 \in \hat{s}_1$ such that $s_1 \rightarrow^\sigma s_2$ for some action sequence σ, in the original transition system.*

Proof. For equalized states $\hat{s}_1, \hat{s}_2$, having $\hat{s}_2 \in \Phi_B(\hat{s}_1)$ means that $\hat{s}_2$ satisfies a goal condition that is determined at $\hat{s}_1$, and is reachable via executing some plan σ. With the assumption that (1) holds, we can apply backwards tracking from any state $s_2 \in \hat{s}_2$ following the transitions Φ corresponding to the actions in the plan σ backwards. In the end, we can find a concrete state $s_1 \in \hat{s}_1$ from which one can reach the state $s_2 \in \hat{s}_2$ by applying the plan σ in the original transition system. □

Thus, we can conclude the following corollary, with the requirement of only having initial states clustered into the equalized initial states (i.e. no "non-initial" state is mapped to an initial equalized state). Technically, it should hold that $\forall s \in S_0 : h^{-1}(h(s)) \subseteq S_0$.

Corollary 1. *If there is a trajectory in the equalized transition system with initial state clustering from an equalized initial state $\hat{s}_0$ to g_∞, then it is possible to find a trajectory in the original transition system from some concrete initial state $s_0 \in \hat{s}_0$ to g_∞.*

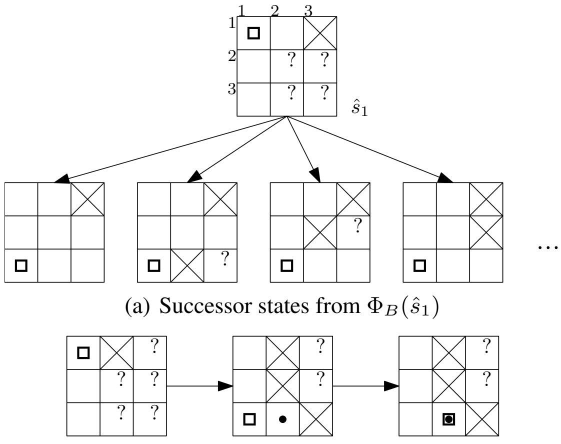
We want to be able to study the reactive policy through the equalized transition system. In case the policy does not work as expected, there should be trajectories that shows the reason of the failure. Knowing that any such trajectory found in the equalized transition system exists in the original transition system is enough to conclude that the policy indeed does not work.

Current assumptions can not avoid the case where a plan σ returned by *Reach* on the equalized transition system does not have a corresponding trajectory in the original transition system. Therefore, we consider an additional condition as

$$\hat{s}' \in \hat{\Phi}(\hat{s}, a) \Leftrightarrow \forall s \in \hat{s},\ \exists s' \in \hat{s}' : s' \in \Phi(s, a) \qquad (2)$$

that strengthens the properness condition (1). Under this condition, every plan returned by *Reach* can be successfully executed in the original transition system and will establish the target g_B. However, still we may lose trajectories of the original system as by clustering states they might not turn into conformant plans. Then one would need to modify the description of determining targets, i.e. the set of targets G_B and the function $\mathcal{B}$.

Example 2. Remember the environment and the policy described in the running example, and consider the scenario shown in Figure 4(a). It shows a part of the equalized transition system constructed according to the policy. The states that are not distinguishable due to the partial observability are clustered into the same state.



(a) Successor states from $\Phi_B(\hat{s}_1)$

(b) Disregarding irrelevant states

Figure 4: Parts of an equalized transition system

The policy is applied according to current observations, and the successor states show the possible resulting states. The aim of the policy is to have the agent move to the farthest reachable point, which for $\hat{s}_1$ is $(3, 1)$. As expected, there can be several states that satisfy the target $g_B{=}robotAt(3, 1)$. The successor states of $\Phi_B(\hat{s}_1)$ is determined by $Res(\hat{s}_1, \sigma)$ computing the possible resulting states after executing the plan σ returned by $Reach(\hat{s}_1, g_B)$. Considering that the agent will gain knowledge about the environment while moving, there are several possibilities for the resulting state.

Notice that this notion of a transition system can help in reducing the number of states, due to the fact that it is able to disregard states with information on fluents that does not have any effect on the system's behavior. For example, Figure 4(b) shows a case where the unknown parts behind the obstacles are not relevant to the agent's behavior, i.e. the person can be found nonetheless.

Relation with Action Languages

In this section, we describe how our definition of a higher-level transition system that models the behavior can fit into the action languages. Given a program defined by an action language and its respective (original) transition system, we now describe how to model this program following a reactive policy and how to construct the corresponding equalized transition system according to the policy.

Classifying the state space

The approach to classify the (original) state space relies on defining a function that classifies the states. There are at least two kinds of such classification; one can classify the states depending on whether they give the same values for certain fluents and omit the knowledge of the values of the remaining fluents, or one can introduce a new set of fluents and classify the states depending on whether they give the same values for the new fluents:

- Type 1: Extend the set of truth values by $\mathbf{V}' = \mathbf{V} \cup \{\mathsf{u}\}$, where u denotes the value to be *unknown*. Extend the value function by $V' : \mathbf{F} \times S \to \mathbf{V}'$. Then, consider a new set of groups of states, $\widehat{S} = \{\hat{s}_1, \ldots, \hat{s}_n\}$, where a group state $\hat{s}_i$ contains all the states $s \in S$ that give the same values for all $p \in \mathbf{F}$, i.e. $\widehat{S} = \{\hat{s} \mid \forall d, e \in S,\ d, e \in \hat{s} \Longleftrightarrow \forall p \in \mathbf{F} : V'(p, d){=}V'(p, e)\ \}$. The value function for the new group of states is $\widehat{V} : \mathbf{F} \times \widehat{S} \to \mathbf{V}'$.
- Type 2: Consider a new set of (auxiliary) fluent names $\mathbf{F}_a$, where each fluent $p \in \mathbf{F}_a$ is *related* with some fluents of $\mathbf{F}$. The relation can be shown with a mapping $\Delta : 2^{\mathbf{F} \times \mathbf{V}} \to \mathbf{F}_a \times \mathbf{V}$. Then, consider a new set of groups of states, $\widehat{S} = \{\hat{s}_1, \ldots, \hat{s}_n\}$, where a group state $\hat{s}_i$ contains all the states $s \in S$ that give the same values for all $p \in \mathbf{F}_a$, i.e. $\widehat{S} = \{\hat{s} \mid \forall d, e \in S,\ d, e \in \hat{s} \Longleftrightarrow \forall p \in \mathbf{F}_a : V(p, d){=}V(p, e)\ \}$. The value function for the new group of states is $\widehat{V} : \mathbf{F}_a \times \widehat{S} \to \mathbf{V}$.

We can consider the states in the same classification to have the same *profile*, and the classification function h as a membership function that assigns the states into groups.

Remarks: (1) In Type 1, introducing the value *unknown* for the fluents allows for describing sensing actions and knowing the true value of a fluent at a later state. Also, one needs to give constraints for a fluent to have the *unknown* value. e.g. it can't be the case that a fluent related to a grid cell is unknown while the robot is able to observe it.
(2) In Type 2, one needs to modify the action descriptions according to the newly defined fluents and define *abstract actions*. However, in Type 1, the modification of the action definitions is not necessary, assuming that the actions are defined in a way that the fluents that are used when determining an action always have *known* values.

Once a set of equalized states is constructed according to the classification function, one needs to define the reactive policy to determine the transitions. Next, we describe how a policy can be defined from an abstract point of view, through a *target language* which figures out the targets and helps in determining the course of actions, and show how the transitions are constructed.

Defining a target language

Let $\widehat{\mathbf{F}}$ denote the set of fluents that the equalized transition system is built upon. Let $\mathcal{F}(\widehat{\mathbf{F}})$ denote the set of formulas in an abstract language that can be constructed over $\widehat{\mathbf{F}}$.

We consider a declarative way of finding targets. Let $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}}){\subseteq}\mathcal{F}(\widehat{\mathbf{F}})$ be the set of formulas that describe target determination. Let $\mathcal{F}_{G_B}(\widehat{\mathbf{F}}){\subseteq}\mathcal{F}(\widehat{\mathbf{F}})$ denote the set of possible targets that can be determined via the evaluation of the formulas $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}})$ over the related fluents in the equalized states.

Notice that separation of the target determining formulas $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}})$ and the targets $\mathcal{F}_{G_B}(\widehat{\mathbf{F}})$ is to allow for outsourced planners that are able to understand simple target formulas. These planners do not need to know about the target language in order to find plans. However, if one is able to use planners that are powerful enough, then the target language can be given as input to the planner, so that the planner determines the target and finds the corresponding plan.

To define a relation between $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}})$ and $\mathcal{F}_{G_B}(\widehat{\mathbf{F}})$, we introduce some placeholder fluents. Let $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}}) = \{f_1, \dots, f_n\}$ be the set of target formulas. Consider a new set of fluents $\widehat{\mathbf{F}}_{\mathcal{B}} = \{p_{f_1}, \dots, p_{f_n}\}$ where each of the formulas in $\mathcal{F}_{\mathcal{B}}$ is represented by some fluent. The value of a fluent depends on whether its respective formula is satisfied or not, i.e. for a state s, $s \models f \iff V(p_f, s) = \mathsf{t}$. Now consider a mapping $\mathcal{M} : 2^{\widehat{\mathbf{F}}_{\mathcal{B}}} \to 2^{\mathcal{F}_{G_B}(\widehat{\mathbf{F}})}$ where

$$\mathcal{M}(\{p_{f_1}, p_{f_2}, \dots, p_{f_m}\}) = \{g_1, \dots, g_r\}, m \le n \text{ and } r \ge 1$$

means that if there is a state s such that $s \models f_i$, $1 \le i \le m$ and $s \not\models f$ for the remaining formulas $f \in \mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}}) \setminus \{f_1, \dots, f_m\}$, then in the successor state s' of s, $s' \models g_i$ for some $1 \le i \le r$, should hold. We consider the output of M to be a set of targets in order to represent the possibility of nondeterminism in choosing a target.

Transition between states

The transition for the equalized transition system can be denoted with $\widehat{R} \subseteq \widehat{S} \times \widehat{S}$, where $\widehat{R}$ corresponds to the policy execution function Φ_B that uses (a) the target language to determine targets, (b) an outsourced planner (corresponding to the function *Reach*) to find conformant plans and (c) the computation of executing the plans (corresponding to the function *Res*). The outsourced planner finds a sequence of actions $\sigma \in 2^{\mathcal{A}}$ from an equalized state $\hat{s}$ to one of its determined targets g_B. Then the successor equalized states are computed by executing the plan from $\hat{s}$. Transition $\widehat{R}$ shows the resulting states after applying the policy.

Example 3. Let us consider a simple blocksworld example where a policy (of two phases) is defined as follows:

- if at phase 1 and not all the blocks are on the table, move one free block on a stack with highest number of blocks to the table.
- if all the blocks are on the table, move to phase 2.
- if at phase 2 and not all the blocks are on top of each other, move one of the free blocks on the table on top of the stack with more than one block (if exists any, otherwise move the block on top of some block).

Since the policy does not take blocks' labels into consideration, a classification can be of the following form for n number of blocks: We introduce an n-tuple $\langle b_1, \dots, b_n \rangle$ to denote equalized states such that for $i \le n$, b_i would represent the number of stacks that have i blocks. For example, for 4 blocks, a state $\langle 1, 0, 1, 0 \rangle$ where $b_1 = 1, b_2 = 0, b_3 = 1, b_4 = 0$ would represent all the states in the original transition system with the profile "contains a stack of 1 block and a stack of 3 blocks". Notice that in the original transition system for 4 labeled blocks, there are 24 possible states that have this profile and if the blocks need to be in order, then there are 4 possible states.

Figure 5 demonstrates the corresponding equalized transition system for the case of 4 blocks. The equalized transition system for this example is in the following form:

- $\widehat{S}$ is the set of equalized states according to the abstraction as described above.

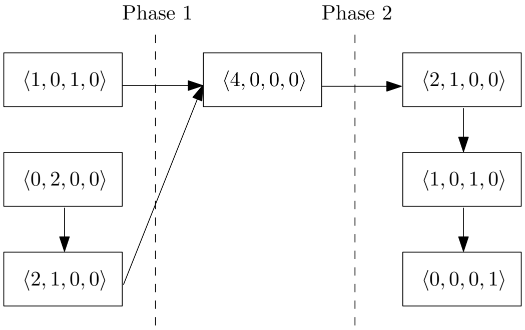


Figure 5: Eq. transition system of blocksworld ($n = 4$)

- $\widehat{S}_0 \in \widehat{S}$ is the initial equalized states (all elements of $\widehat{S}$ except $\langle 0, \dots, 0, 1 \rangle$).
- $G_B = \widehat{S}$, since the policy is related with all the blocks, it can determine targets as the whole states.
- $\mathcal{B} : \widehat{S} \to \widehat{S}$ is the target function.
- $\Phi_B : \widehat{S} \to \widehat{S}$ is the policy execution function, returning the resulting successor state after applying one action desired by the behavior, shown as in Figure 5.

Application on Action Language $\mathcal{C}$

In this section, we describe how one can construct an equalized transition system for a reactive system that is represented using the action language $\mathcal{C}$ (Giunchiglia and Lifschitz 1998). First, we give some background information about the language $\mathcal{C}$, then move on to the application of our definitions.

Syntax A *formula* is a propositional combination of fluents.

Given a propositional action signature $\langle \{\mathsf{f}, \mathsf{t}\}, \mathbf{F}, \mathbf{E} \rangle$, whose set $\mathbf{E}$ of elementary action names is disjoint from $\mathbf{F}$, an *action description* is a set of expressions of the following forms:

- *static laws:*

$$\textbf{caused } F \textbf{ if } G, \tag{3}$$

  where F and G are formulas that do not contain elementary actions;
- *dynamic laws:*

$$\textbf{caused } F \textbf{ if } G \textbf{ after } U, \tag{4}$$

  where F and G are as above, and U is a formula.

Semantics The transition system $\langle S, V, R \rangle$ *described* by an action description D is defined as follows:

(i) S is the set of all interpretations s of $\mathbf{F}$ such that, for every static law (3) s satisfies F if s satisfies G,

(ii) $V(P, s) = s(P)$, i.e. identify s with $V(P, s)$,

(iii) R is the set of all triples $\langle s, A, s' \rangle$, $A \subseteq \mathbf{E}$, such that s' is the only interpretation of $\mathbf{F}$ which satisfies the heads of all

- static laws (3) in D for which s' satisfies G, and

- dynamic laws (4) in D for which s' satisfies G and $s \cup A$ satisfies U.

We focus on a fragment of the language $\mathcal{C}$ where the heads of the static and dynamic laws only consist of literals. This restriction on the laws reduces the cost of evaluating the transitions $\langle s, A, s' \rangle \in R$ to polynomial time. Thus, we match the conditions on complexity from above. Furthermore, by well-known results on the complexity of action language $\mathcal{C}$ (Turner 2002; Eiter et al. 2004) all the results in Theorems 1-5 can be turned into completeness results already for this fragment.

Defining a policy

Let $\widehat{\mathbf{F}}$ be the set of fluents that are relevant to the policy. The target language is defined explicitly via static laws using the fluents in $\widehat{\mathbf{F}}$, denoted $\mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}})$, where a target is determined by the evaluation of these formulas in a state.

Example 4. An example of a target language for the running example uses causal laws from $\mathcal{C}$:

$$
\begin{aligned}
&\textbf{caused } \mathit{target}(X1, Y1) \textbf{ if}\\
&\qquad\qquad \mathit{robotAt}(X, Y) \wedge \mathit{farthest}(X, Y, X1, Y1)\\
&\qquad\qquad\qquad\qquad \wedge\ \mathit{not\ personDetected}.\\
&\textbf{caused } \mathit{personDetected} \textbf{ if } \mathit{personDetected}(X, Y).\\
&\textbf{caused } \mathit{targetPerson}(X, Y) \textbf{ if } \mathit{personDetected}(X, Y).\\
&\textbf{caused } \mathit{personFound} \textbf{ if } \mathit{personDetected}(X, Y)\\
&\qquad\qquad\qquad\qquad \wedge\ \mathit{robotAt}(X, Y).
\end{aligned}
$$

where $\mathcal{F}_{G_B}(\widehat{\mathbf{F}})$ consists of all atoms $\mathit{target}(X, Y)$ and $\mathit{targetPerson}(X, Y)$ for $1 \leq X \leq n, 1 \leq Y \leq n$.

The target of a state according to the policy is computed through joint evaluation of these causal laws over the state with the *known* fluents about the agent's location and the reachable points. Then, the outsourced planner may take as input the agent's current location and the target location, to find a plan to reach the target.

Equalized transition system The equalized transition system $\langle \widehat{S}, \widehat{V}, \widehat{R} \rangle$ that describes the policy is defined as follows:

(i) $\widehat{S}$ is the set of all interpretations of $\widehat{\mathbf{F}}$ such that, for every static law (3) $\hat{s}$ satisfies F if $\hat{s}$ satisfies G,

(ii) $\widehat{V}(P, \hat{s}) = \hat{s}(P)$, where $P \in \widehat{\mathbf{F}}$,

(iii) $\widehat{R} \subseteq \widehat{S} \times \widehat{S}$ is the set of all $\langle \hat{s}, \hat{s}' \rangle$ such that

  (a) for every $s' \in \hat{s}'$ there is a trajectory from some $s \in \hat{s}$ of the form $s, A_1, s_1, \ldots, A_n, s'$ in the original transition system;

  (b) for static laws $f_1, f_2, \ldots, f_m \in \mathcal{F}_{\mathcal{B}}(\widehat{\mathbf{F}})$ for which $\hat{s}$ satisfies the body, it holds that $\hat{s}' \models g$ for some $g \in \mathcal{M}(p_{f_1}, \ldots, p_{f_m})$.

Notice that in the definition of the transition relation $\widehat{R}$ in (iii) there is no description of (a) how a trajectory is computed or (b) how a target is determined. This gives flexibility on the implementation of these components.

Other languages can be similarly used to describe the equalized transition system, as long as they are powerful enough to express the concepts from the previous section.

Related Work

There are works being conducted on the verification of GOLOG programs (Levesque et al. 1997), a family of high-level action programming languages defined on top of action theories expressed in the situation calculus. The method of verifying properties of non-terminal processes are sound, but do not have the guarantee of termination due to the verification problem being undecidable (De Giacomo, Ternovskaia, and Reiter 1997; Claßen and Lakemeyer 2008). By resorting to action formalisms based on description logic, decidability can be achieved (Baader and Zarrieß 2013).

Verifying temporal properties of dynamic systems in the context of data management is studied by (Calvanese et al. 2013) in the presence of description logic knowledge bases. However, target establishment and planning components are not considered in these works, and they do not address real life environment settings.

The logical framework for agent theory developed by Rao and Georgeff (1991) is based on beliefs, desires and intentions, in which agents are viewed as being rational and acting in accordance with their beliefs and goals. There are many different agent programming languages and platforms based on the BDI approach. Some works carried out on verifying properties of agents represented in these languages, such as (Bordini et al. 2006; Dennis et al. 2012). These approaches consider very complex architectures that even contain a plan library where plans are matched with the intentions or the agent's state and manipulate the intentions.

Synthesizing and Verifying Plans There have been various works on synthesizing plans via symbolic model checking techniques by Cimatti et al. (1998b; 1998a), Bertoli et al. (2006). These approaches are able to solve difficult planning problems like strong planning and strong cyclic planning.

Son and Baral (2001) extend the action description language by allowing sensing actions and allow to query conditional plans. These conditional plans are general plans that consist of sensing actions and conditional statements.

These works address a different problem then ours. When nondeterminism and partial observability are taken into account, finding a plan that satisfies the desired results in the environment is highly demanding. We consider a much less ambitious approach where given a behavior, we aim to check whether or not this behavior gives the desired results in the environment. However, our framework is capable emulating the plans found by these works.

Execution Monitoring There are logic-based monitoring frameworks that monitor the plan execution and recover the plans in case of failure. The approaches that are studied are replanning (De Giacomo, Reiter, and Soutchanski 1998), backtracking to the point of failure and continuing from there (Soutchanski 2003), or diagnosing the failure and recovering from the failure situation (Fichtner, Großmann, and Thielscher 2003; Eiter et al. 2007).

These works consider the execution of a given plan, while we consider a given reactive policy that determines targets and use (online) planning to reach these targets.

Conclusion and Future Work

In this paper, we described a high-level representation that models reactive behaviors, and integrates target development and online planning capabilities. Flexibility in these components does not bound one to only use action languages, but allows for the use of other formalizations as well. For future work, one could imagine targets to depend on further parameters or to incorporate learning from experience in the framework. It is also possible to use other plans, e.g. short conditional plans, in the planner component.

The long-term goal of this work is to check and verify properties of the reactive policies for action languages. In order to solve these problems practically, it is necessary to use techniques from model checking, such as abstraction, compositional reasoning and parameterization. Also, the use of temporal logic formulas is needed to express complex goals such as properties of the policies. Our main target is to work with action languages, and to incorporate their syntax and semantics with such model checking techniques. The general structure of our framework allows one to focus on action languages, and to investigate how to merge these techniques.

References

Baader, F., and Zarrieß, B. 2013. Verification of Golog programs over description logic actions. *Frontiers of Combining Systems* 181–196.

Bertoli, P.; Cimatti, A.; Riveri, M.; and Traverso, P. 2006. Strong planning under partial observability. *Artificial Intelligence* 170(4):337–384.

Bordini, R. H.; Fisher, M.; Visser, W.; and Wooldridge, M. 2006. Verifying multi-agent programs by model checking. *Autonomous agents and multi-agent systems* 12(2):239–256.

Calvanese, D.; De Giacomo, G.; Montali, M.; and Patrizi, F. 2013. Verification and synthesis in description logic based dynamic systems. In *Web Reasoning and Rule Systems*. Springer. 50–64.

Cimatti, A.; Riveri, M.; and Traverso, P. 1998a. Automatic OBDD-based generation of universal plans in non-deterministic domains. In *Proc. of AAAI/IAAI*, 875–881.

Cimatti, A.; Riveri, M.; and Traverso, P. 1998b. Strong planning in non-deterministic domains via model checking. *AIPS* 98:36–43.

Claßen, J., and Lakemeyer, G. 2008. A logic for non-terminating Golog programs. In *Proc. of KR*, 589–599.

De Giacomo, G.; Reiter, R.; and Soutchanski, M. 1998. Execution monitoring of high-level robot programs. In *Proc. of KR*, 453–465.

De Giacomo, G.; Ternovskaia, E.; and Reiter, R. 1997. Non-terminating processes in the situation calculus. In *Working Notes of Robots, Softbots, Immobots: Theories of Action, Planning and Control, AAAI97 Workshop*.

Dennis, L. A.; Fisher, M.; Webster, M. P.; and Bordini, R. H. 2012. Model checking agent programming languages. *Automated Software Engineering* 19(1):5–63.

Eiter, T.; Faber, W.; Leone, N.; Pfeifer, G.; and Polleres, A. 2004. A logic programming approach to knowledge-state planning: Semantics and complexity. *ACM Trans. Comput. Log.* 5(2):206–263.

Eiter, T.; Ianni, G.; Schindlauer, R.; and Tompits, H. 2005. A Uniform Integration of Higher-Order Reasoning and External Evaluations in Answer-Set Programming. In *Proc. of IJCAI*, 90–96.

Eiter, T.; Erdem, E.; Faber, W.; and Senko, J. 2007. A logic-based approach to finding explanations for discrepancies in optimistic plan execution. *Fundamenta Informaticae* 79(1-2):25–69.

Fichtner, M.; Großmann, A.; and Thielscher, M. 2003. Intelligent execution monitoring in dynamic environments. *Fundamenta Informaticae* 57(2-4):371–392.

Fink, M.; Germano, S.; Ianni, G.; Redl, C.; and Schüller, P. 2013. Acthex: Implementing HEX programs with action atoms. *Logic Programming and Nonmonotonic Reasoning* 317–322.

Gebser, M.; Grote, T.; and Schaub, T. 2010. Coala: A compiler from action languages to ASP. In *Proc. of JELIA*, 360–364. Springer Heidelberg.

Gelfond, M., and Lifschitz, V. 1993. Representing action and change by logic programs. *The Journal of Logic Programming* 17(2):301–321.

Gelfond, M., and Lifschitz, V. 1998. Action languages. *Electronic Transactions on AI* 3(16).

Giunchiglia, E., and Lifschitz, V. 1998. An action language based on causal explanation: Preliminary report. In *Proc. of AAAI/IAAI*, 623–630.

Giunchiglia, E.; Lee, J.; Lifschitz, V.; McCain, N.; and Turner, H. 2004. Nonmonotonic causal theories. *Artificial Intelligence* 153(1):49–104.

Kowalski, R. A., and Sadri, F. 1999. From logic programming towards multi-agent systems. *Ann. Math. Artif. Intell.* 25(3-4):391–419.

Levesque, H. J.; Reiter, R.; Lesperance, Y.; Lin, F.; and Scherl, R. B. 1997. GOLOG: A logic programming language for dynamic domains. *The Journal of Logic Programming* 31(1):59–83.

Lifschitz, V. 1999. Action languages, answer sets and planning. In *The Logic Programming Paradigm: a 25-Year Perspective*, 357–373. Springer.

Lifschitz, V. 2008. What is answer set programming? In *Proc. of. AAAI*, 1594–1597.

Rao, A. S., and Georgeff, M. P. 1991. Modeling rational agents within a BDI-architecture. In *Proc. of KR*, 473–484.

Son, T. C., and Baral, C. 2001. Formalizing sensing actions – a transition function based approach. *Artificial Intelligence* 125(1):19–91.

Soutchanski, M. 2003. High-level robot programming and program execution. In *Proc. of ICAPS Workshop on Plan Execution*.

Turner, H. 2002. Polynomial-length planning spans the polynomial hierarchy. In *Proc. of JELIA*, 111–124. Springer.

Static and Dynamic Views on the Algebra of Modular Systems

Eugenia Ternovska

Abstract

The paper develops a knowledge representation framework called Algebra of Modular Systems. Each module can be given by a knowledge base, be an agent, a knowledge base, an ASP, ILP, CP program, etc.

Under the "still" or "static" view, the algebra is the same as Codd's relational algebra (with recursion added), but operations are applied to classes of structures instead of relational tables. Under the "dynamic" view, when we indicate the direction of information flow, the same algebra is a modal temporal logic.

We use the algebra for a high-level encoding of problem solving on graphs using Dynamic Programming on tree decompositions. We also use it to specify an algorithm for solving quantified boolean formulas.

We show that the well-known Propositional Dynamic Logic is a fragment of the algebra with information flow.

We demonstrate a connection of our formalism to the situation calculus with the cognitive robotics language Golog.

Introduction

> *"There is a point where in the mystery of existence contradictions meet; where movement is not all movement and stillness is not all stillness... where the idea and the form, the within and the without, are united; where infinite becomes finite, yet not losing its infinity."* *Rabindranath Tagore*

In 1970 Edgar (Ted) F. Codd introduced a relational data model and two query languages: relational calculus and relational algebra. Relational calculus is what we usually call FO logic. The key contribution of Codd was to associate with a declarative specification language (FO logic), a procedural counterpart which is the relational algebra, that later was implemented by smart engineers and became a multi-billion dollar industry of relational database management systems (RDBMS).

A significant change has happened in the past decade. While at the low level everything boils down to SQL queries, interactions between "larger pieces" became increasingly important. Such "larger pieces"are business enterprises, web services, software, solvers in the world of declarative problem solving, etc. The emergence of large-scale interactions poses significant challenges to KR and database researchers. Among those challenges, the most urgent is *scallability of proposed approaches to data management*. The importance of scallability is justified by the Moore's Law (1965), according to which the number of transistors in a dense integrated circuit doubles approximately every two years. Moore's prediction proved accurate for several decades: capabilities of hardware grow exponentially.

We expect that, as hardware capabilities grow, people will be able to combine, as easily as relational tables in SQL queries, entities of a completely different magnitude:

- *knowlege bases*, e.g. representing the work of various enterprises, into complex business processes,
- *specifications of combinatorial optimization and search problems* for solving new, more complex, problems.

To be more concrete, let us think that each one of these heterogeneous "larger pieces" is represented declaratively, e.g., by Logic Programming rules, ILP equations, SMT theories, FO knowledge bases, etc.

While database queries, expressed using Codd's relational calculus, can be viewed as *relations definable with respect to a structure* (a database), declarative problem specifications can be understood as *axiomatizations of classes of structures*. The two notions (in italic) are defined in two consecutive chapters in the classic textbook on mathematical logic (Enderton 1972).

The main idea of our Algebra of Modular Systems is to lift Codd's algebra from operations on relational tables to operations on *classes of structures* and to add *recursion*.[1]

The operations of the algebra (to be defined formally) can be seen in this grammar for Modular Systems:

$$E ::= \bot | M_i | Z_j | E \times E | E + E | - E | \pi_\delta E | \sigma_\Theta E | \mu Z_j . E,$$

and they roughly correspond to conjunction, disjunction, negation, existential quantifier (π_δ) of FO logic. Selection (σ_Θ) restricts the module to structures satisfying the formula Θ. In addition, least fixed point is used. We now consider two examples, a simple and a more realistic.

Example 1. Let $M_{\rm HC}(V, X, Y)$ and $M_{\rm 2Col}(V, X, Z, T)$ be atomic modules "computing" a Hamiltonian Circuit, and a 2-colouring. They can do it in different ways. For example, $M_{\rm HC}$ can use ASP, and $M_{\rm 2Col}$ an imperative program or a

[1] Here, we require a model-theoretic semantics, although various generalizations are possible.

human child with two pencils. Here, V is a relational variable of arity 1, X, Y are relational variables of arity 2, and the first module decides if Y forms a Hamiltonian Circuit (represented as a set of edges) in the graph given by vertex set V and edge set X. Variable X of the second module has arity 2, and variables Z, T are unary; the module decides if unary relations Z, T specify a proper 2-colouring of the graph with edge set X. The following algebraic expression determines a combination of 2-Colouring and Hamiltonian Circuit, that is whether or not there is a 2-colourable Hamiltonian Circuit.[2]

$$\begin{aligned} & M_{\mathrm{2Col-HC}}(V, X, Z, T) := \\ & \pi_{V,X,Z,T}((M_{\mathrm{HC}}(V, X, Y) \times M_{\mathrm{2Col}}(V, Y, Z, T)). \end{aligned} \tag{1}$$

Projection "keeps" V, X, Z, T and hides the interpretation of Y in M_{HC}, since it is the same as Y's in M_{2Col}.

Example 2. This modular system can be used by a company that provides logistics services (arguments of atomic modules are omitted).

$$M_{LSP} := \sigma_{B \equiv B'}(M_K \times M_{TSP}).$$

It decides how to pack goods and deliver them. It solves two NP-complete tasks interactively, – Multiple Knapsack (module M_K) and Travelling Salesman Problem (module M_{TSP}). The system takes orders from customers (items to deliver, their profits, weights), and the capacity of trucks, decides how to pack items in trucks, and for each truck, solves a TSP problem. The feedback B' about solvability of TSP is sent back to M_K. The two sub-problems, M_K and M_{TSP}, are solved by different sub-divisions of the company (with their own business secrets) that cooperate towards the common goal. A solution to the compound module, M_{LSP}, to be acceptable, must satisfy both sub-systems.

Most practical examples use simple combinations of modules, where only conjunctions, disjunctions, perhaps with projections and selections, are used. The university timetabling example (Järvisalo et al. 2009) and Examples 1, 2 are of this kind. Examples where a (declaratively specified) module calls itself recursively are much harder to come by. We will however show two natural applications. In the first one, combinatorial search problems on graphs are solved using their tree decompositions. In that application, the algebra allows us to write compact specifications of Dynamic Programming algorithms on tree decompositions. In the second application, our algebra is used to describe a recursive algorithm for solving quantified boolean formulas (QBFs). Note that here we are talking about recursion *over a module*, not over a predicate symbol, which happens frequently inside, say, ASP modules. Recursion over predicate symbols is not really needed in the algebra itself since free second-order variables are implicitly $\exists$SO-quantified, which gives at least as much expressive power as such a recursion can provide. Feedbacks from module to module, however, (as in Example 2) are very useful.

Dynamic View Problem solving often involves finding solutions for given inputs. Most combinatorial problems are of that form. The Logistics Service Provider in Example 2 has, e.g., customer requests as an input, and routes and packing solutions as outputs. One can have e.g. edges of a graph on the input to formula (1), and colours on the output. To capture this meaning, we introduce information flow to the algebra. Thus, we can now reason about actions or changes performed by the modules. Interpreting our algebra over a transition system gives rise to a modal temporal logic.

There are several examples of connections between classical ("still" or "static") and modal ("dynamic") logics, and the notion of bisimulation plays an important role there. These are the so-called *model-theoretic characterization theorems*. For example, van Benthem shown that modal logic is a bisimulation-invariant fragment of first-order logic. Janin and Walukiewicz proved that the modal μ-calculus $L\mu$ is the bisimulation-invariant fragment of monadic second-order logic MSO. (Abu Zaid, Grädel, and Jaax 2014) studied a related notion of bisimulation safety introduced by van Benthem. They introduced a new logic called Bisimulation Safe Fixed Point Logic (BSFP) that is more expressive than MSO. The BSFP logic is a very elegant formalism that uses both unary and binary fixed points. We will discuss it in more detail and use this logic in the context of modular systems with information flow. We investigate connections with other modal temporal logics. From the connection to BSFL, it follow that Propositional Dynamic Logic (PDL) is a fragment of our algebra with information flow.

We were quite surprised to notice that essentially the same constructions as in BSFP, but in axiomatic setting, appeared some 18 years earlier in the context of the situation calculus and GOLOG (Levesque et al. 1997; De Giacomo, Ternovskaia, and Reiter 1997). We discuss connections to the situation calculus in detail.

The duality of the "static" and "dynamic" views on modular systems opens new possibilities for developing algorithms to answer questions about the "static" algebra. For example, finding solutions to multi-language constraint problems can be done by solving Model Checking task for the modal temporal logic. This implies that both SAT-based and symbolic model checking techniques can be used.

Preliminaries A *vocabulary* (denoted, e.g. $\tau, \sigma, \varepsilon, \nu$) is a finite sequence of non-logical (predicate and function) symbols, each with an associated arity. A τ-structure, e.g. $\mathcal{A} = (A; S_1^{\mathcal{A}}, ..., S_n^{\mathcal{A}}, f_1^{\mathcal{A}}, ..., f_m^{\mathcal{A}}, c_1^{\mathcal{A}}, ..., c_l^{\mathcal{A}})$ is a domain A together with interpretations of predicate symbols, function and constants (0-ary functions) in τ. To simplify presentation, we view functions as a particular kind of relations and consider relational structures only. We use notations $vocab(\mathcal{A})$, $vocab(\phi)$, $vocab(M)$ to denote vocabulary of structure $\mathcal{A}$, formula ϕ and module M, respectively, and we use $\mathcal{B}|_\sigma$ to mean structure $\mathcal{B}$ restricted to vocabulary σ. Symbol $:=$ means "denotes" or "is by definition".

Least Fixed Point logic FO(LFP) is broadly used in CS and is described in several books. For a background on that logic we refer to, e.g. (Grädel et al. 2007).

Just as the basic modal logic is a fragment of first-order logic, the modal mu-calculus $L\mu$ (that includes the well-known temporal logics CTL, CTL* and LTL) is a frag-

[2] We use $:=$ for "is by definition".

ment of FO(LFP). Moreover, the modal mu-calculus $L\mu$ is a bisimulation-invariant fragment of monadic second-order logic MSO. More details on fixed point logics and MSO can be found in (Dawar and Gurevich 2002; Libkin 2004; Grädel et al. 2007).

"Still" Algebra

We call this version of the algebra "still" to distinguish it from the version with information propagation below.[3]

Syntax Let τ be a fixed vocabulary. W.l.o.g. we assume that τ is relational (it does not contain function symbols). Let $\tau_M = \{M_1, M_2, \dots\}$ be a fixed vocabulary of *atomic module symbols*, $\tau \cap \tau_M = \varnothing$. Atomic module symbols are of the form $M_i(X_{i_1}, \dots, X_{i_k})$ (also written $M_i(\overline{X})$), where each X_i is a relational variable. Each X_j has an associated arity a_j. The set $\{X_{i_1}, \dots, X_{i_k}\}$ is called the *variable vocabulary of* M_i and is denoted $vvoc(M_i)$. We also allow relational constants from τ in place of the variables, provided their arities match. In this case, $vocab(M_i)$ denotes the combined (variable and constant) vocabularies of M_i. Let Z_1, Z_2, ... be a collection of *module variables*. Algebraic expressions for *modules* are built by the grammar:

$$E ::= \bot | M_i | Z_j | E \times E | E + E | - E | \pi_\delta E | \sigma_\Theta E | \mu Z_j.E. \quad (2)$$

Modules in τ_M are *atomic*. Modules that are not atomic are called *compound*. The operations (except $\mu Z_j.E$) are like in Codd's relational algebra, but are of a higher order,[4] and are defined on *classes of structures* rather than on relational tables. The three set-theoretic operations are union ($+$), intersection ($\times$), complementation ($-$). Projection ($\pi_\delta E$) is a family of unary operations, one for each δ. Each relational symbol (constant or variable) in δ must appear in E. The operation restricts each structure $\mathcal{A}$ of M to $\mathcal{A}|_\delta$ leaving the interpretation of other symbols open. Thus, it *increases* the number of models. The condition Θ in selection $\sigma_\Theta E$ is an expression of the form $L_1 \equiv L_2$, where L_i is a relational variable or a relational constant from τ, or 'R', where R is a relation (set of tuples of domain elements).[5] Thus, we bring semantic elements into syntax in the latter case. Selection *reduces* the number of models. [6]

Semantics To interpret algebraic expressions, we use valuations $(v, \mathcal{V})$. Intuitively, v maps relational variables in $vvoc(M_i)$ to symbols from a relational vocabulary τ so that the arities of the relational variables in $vvoc(M_i)$ match those of the corresponding symbols in τ. Function $\mathcal{V}$ is parameterised by v and provides a domain (which does not have to be finite), and interpretations of atomic modules M_i as explained below. Let $\mathcal{C}$ be the set of all τ-structures over the domain fixed by $\mathcal{V}$. Valuation $\mathcal{V}$ maps each atomic module symbol M_i to a subset $\mathcal{V}(v, M_i)$ of $\mathcal{C}$ so that for any two τ-structures $\mathcal{A}_1$, $\mathcal{A}_2$ which coincide on $vocab(M_i)$, we have $\mathcal{A}_1 \in \mathcal{V}(v, M_i)$ iff $\mathcal{A}_2 \in \mathcal{V}(v, M_i)$. All of the above applies to module variables Z_j as well. In practice, $\mathcal{V}$ can, for example, associate one module symbols with stable models of an ASP program, another module symbol with models of an ILP encoding, yet another one with a set of databases used by a particular enterprise, etc.

Remark 1. *Note that v resembles a "call by reference" in programming. Valuations $\mathcal{V}$ (parameterized with v) can be viewed as "oracles" or decision procedures associated with modules, and can be of arbitrary computational complexity.*

The *extensions* $[\![E]\!]^{\mathcal{V},v}$ of algebraic expressions E are subsets of $\mathcal{C}$ defined as follows.

$[\![\bot]\!]^{\mathcal{V},v} := \varnothing.$
$[\![M_i]\!]^{\mathcal{V},v} := \mathcal{V}(v, M_i)$ for some v.
$[\![Z_j]\!]^{\mathcal{V},v} := \mathcal{V}(v, Z_j)$ for some v.
$[\![E_1 + E_2]\!]^{\mathcal{V},v} := [\![E_1]\!]^{\mathcal{V},v} \cup [\![E_2]\!]^{\mathcal{V},v}.$
$[\![-E]\!]^{\mathcal{V},v} := \mathcal{C} \setminus [\![E]\!]^{\mathcal{V},v}.$
$[\![\pi_\delta(E)]\!]^{\mathcal{V},v} := \{\mathcal{A} \mid \exists \mathcal{A}' \ (\mathcal{A}' \in [\![E]\!]^{\mathcal{V},v}$ and $\mathcal{A}|_\delta = \mathcal{A}'|_\delta\ )\}.$
$[\![\sigma_{L_1 \equiv L_2} E]\!]^{\mathcal{V},v} := \{\mathcal{A} \mid [\![E]\!]^{\mathcal{V},v}$ and $L_1^{\mathcal{A}} = L_2^{\mathcal{A}}\}.$
$[\![\mu Z_j.E]\!]^{\mathcal{V},v} := \bigcap \{\mathcal{E} \subseteq \mathcal{C} \mid [\![E]\!]^{\mathcal{V}[Z:=\mathcal{E}],v} \subseteq \mathcal{E}\}.$

Here, $\mathcal{V}[Z{:=}\mathcal{E}]$ means a valuation that is exactly like $\mathcal{V}$ except Z is interpreted as $\mathcal{E}$.
Given a well-formed algebraic expression E defined by (2), we say that structure $\mathcal{A}$ *satisfies* E under valuation $(\mathcal{V}, v)$, notation $\mathcal{A} \models_{(\mathcal{V},v)} E$ if $\mathcal{A} \in [\![E]\!]^{\mathcal{V},v}$.
Some useful operations on modules are: Extending the vocabulary of E to a bigger vocabulary: $\pi_\delta(E \times \top)$. Renaming P to Q in E: $\pi_{vocab(E) \setminus \{P\} \cup \{Q\}} \sigma_{P \equiv Q}(E \times \top)$. Difference: $E_1 - E_2 = E_1 \times (-E_2)$. Universal module: $\top = -\bot$. Also, as we would expect, $E_1 \times E_2 = -((-E_1) + (-E_2))$.

Remark 2. *Note that while individual modules are already capable of solving optimization tasks (the optimum value can be given as an output in one of the arguments), the least fixed point construct can generate the least value over a collection of modules combined in an algebraic expression.*

Proposition 1 (Logic Counterpart)**.** *The algebraic operations are equivalently representable in logic, where '$+$' corresponds to disjunction, '$-$' to negation, 'π_ν' to second-order existential quantification over $\tau \setminus \nu$, 'σ_Θ' to conjunction with Θ, $\mu Z.E$ to the least fixed point construct.*

Thus, our formalism is FO(LFP) over modules that are of an arbitrary expressive power.

Example 3. *Expression (1) for $M_{\mathrm{2Col-HC}}(V, X, Z, T)$ is represented in logic as*

$$\exists Y [M_{\mathrm{HC}}(V, X, Y) \wedge M_{\mathrm{2Col}}(V, Y, Z, T)].$$

Dynamic Programming on Tree Decompositions

We use our algebra to specify high-level recursive control in solving combinatorially hard problems using dynamic programming and tree decomposition, along the lines of (Abseher et al. 2014; Charwat and Woltran 2015).

[3] We use the terms "still" and "static" interchangeably.

[4] For example, projection is onto a set of *relational* constants or variables rather than object constants or variables.

[5] A more general version allows Θ to be built up using $\wedge$, $\vee$, $\neg$, from equivalence operators $\equiv$, $\not\equiv$. That choice for Θ may be more efficient computationally, but does not add expressive power since the same effect is achievable trough the other operations.

[6] Selection can be used, in particular, to connect modules by equating relational symbols of equal arity, and to express *grounding* by brining relations over domain elements into the syntax.

Definition 1. *(Robertson and Seymour 1984) A tree decomposition of an undirected graph $G = (V; E)$ is a pair $(\mathcal{T}, \mathcal{X})$ where $\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T})$ is a tree and $\mathcal{X} : V_\mathcal{T} \to 2^V$ assigns to every node $V_\mathcal{T}$ of the tree a set of vertices V from the original graph. The sets of vertices $\mathcal{X} = (X_t)_{t \in V_\mathcal{T}}$ have to satisfy the following conditions: (1) $\bigcup_{t \in V_\mathcal{T}} X_t = V$. (2) $\{x, y\} \in E \Rightarrow \exists t \in V_\mathcal{T} : \{x, y\} \subseteq X_t$. (3) $x \in X_{t'} \wedge x \in X_{t''} \wedge t''' \in path(t', t'') \Rightarrow x \in X_{t'''}$. X_t is also called the bag for the vertex $t \in V_\mathcal{T}$. The width w of the decomposition is $max_{t \in V_\mathcal{T}} |X_t| - 1$. The tree-width k of a graph is the minimum width over all its tree decompositions.*

It follows that every vertex of the graph is contained in some bag of the tree decomposition, adjacent vertices appear together in some bag, and nodes that contain the same vertex are connected. We denote an edge between vertices x, y by $\{x, y\}$. For a decomposition node t, we denote by $E_t := \{\{x, y\} \in E \mid x, y \in X_t\}$ the edges of G induced by the vertices X_t. We focus on a special type of tree decomposition.

Definition 2. *A tree decomposition $\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T})$ is called normalized if each $t \in V_\mathcal{T}$ is of one of the following types: (1) Leaf node: t has no child nodes. (2) Introduction node: t has exactly one child node t' with $X_{t'} \subset X_t$ and $|X_{t'}| = |X_t| - 1$. (3) Removal node: t has exactly one child node t' with $X_t \subset X_{t'}$ and $|X_{t'}| = |X_t| + 1$. (4) Join node: t has exactly two child nodes t' and t'' with $X_t = X_{t'} = X_{t''}$.*

Example 4. Modules used: tree decomposition $M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}, L_t)$ and 3-Colouring $M_{\text{3Col}}(X_t, E_t, R, B, G)$. Intuitively, (V, E) is the original graph, (X_t, E_t) is the current bag with its adges, $X_{t'}, X_{t''}$ are its children, U_t is true on the vertex that has been introduced or removed, and EU_t on the edges that lead to such vertices, and L_t specifies the label in $\{l, j, r, i\}$ of a particular node in tree decomposition. In M_{3Col}, (X_t, E_t) is the graph which we colour with R, B, G. The problem is represented as

$$M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}) \wedge \mu Z.\Psi(Z, M_{\text{TD}}, M_{\text{3Col}}),$$

where Z is a module variable of the form $Z(V, E, X_t, E_t, R, B, G)$ over which recursive iteration is performed. $\mu Z.\Psi(Z, M_{\text{TD}}, M_{\text{3Col}})$ specifies dynamic programming algorithm by recursion over tree decomposition, with $\Psi(Z, M_{\text{TD}}, M_{\text{3Col}}) := \phi_l \vee \phi_i \vee \phi_r \vee \phi_j$.

Base case, leaf:

$$\phi_l := \sigma_{(L_t \equiv `\langle l \rangle' \wedge X_{t'} \equiv \bot \wedge X_{t''} \equiv \bot \wedge U_t \equiv \bot)}[M_{\text{3Col}}(X_t, E_t, R, B, G) \\ \wedge M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}, L_t)]$$

We use $\bot$ for predicate constant "false". The selection above requires that the label of the leaf is l, there are no child nodes (bags) $(X_{t'} \equiv \bot \wedge X_{t''} \equiv \bot)$, and there is no introduced/removed vertex v in the bag X_t $(U_t \equiv \bot)$. Module M_{3Col} performs 3-Colouring of the bag X_t with edges E_t, where the tree decomposition is M_{TD}.
Introduced vertex case:

$$\phi_i := \sigma_{(L_t \equiv `\langle i \rangle' \wedge E_t \equiv EU_t \wedge X_{t''} \equiv \bot)}[\\ Z(V, E, X_{t'}, E_{t'}, R, B, G) \wedge M_{\text{3Col}}(X_t, E_t, R, B, G) \\ \wedge M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}, L_t)]$$

Removed vertex case:

$$\phi_r := \exists R \exists B \exists G[\sigma_{((R \equiv U_t \vee B \equiv U_t \vee G \equiv U_t) \wedge L_t \equiv `\langle r \rangle' \wedge X_{t''} \equiv \bot)}[\\ Z(V, E, X_{t'}, E_{t'}, R, B, G) \wedge M_{\text{3Col}}(X_t, E_t, R, B, G) \\ \wedge M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}, L_t)]$$

Join vertex case:

$$\phi_r := \sigma_{(L_t \equiv `\langle j \rangle')}[\\ Z(V, E, X_{t'}, E_{t'}, R, B, G) \wedge Z(V, E, X_{t''}, E_{t''}, R, B, G) \\ \wedge M_{\text{TD}}(V, E, X_t, E_t, U_t, EU_t, X_{t'}, X_{t''}, L_t)]$$

Dynamic programming algorithms for different problems on tree decompositions of graphs (e.g. from (Charwat and Woltran 2015)) can be formulated in our algebra in a symilar manner.

Algebra with Information Flow

Modules that have *inputs* and *outputs* are very common. Many software programs and hardware devices are of that form. In the Logistics Service Provider (Example 2), e.g. users' requests could be on the input, and truck routes and packing solutions on the output.
In this section, we add information propagation to the algebra, so that modules become binary higher-order input-output relations. This version of the algebra may be called "dynamic". In algebraic expressions and corresponding logic formulas, we underline designated input symbols, i.e., those in σ_M. Output symbols are free (are not quantified). For example:

$$\exists Y[M_{\text{HC}}(\underline{V}, \underline{X}, Y) \wedge M_{\text{2Col}}(\underline{V}, Y, Z, T)]. \qquad (3)$$

The quantified symbol Y is not visible from the outside. The output vocabulary of this compound modular system is $\varepsilon = \{Z, T\}$ (for the two colours), the input vocabulary is $\sigma = \{V, X\}$ (for the vertices and edges). In fact *any* direction of information propagation can be specified, e.g. from colours to graphs in 2-Colouring.
Fixing an input and an output vocabularies in some modules allows us to talk about the *model expansion (MX)* task (Mitchell and Ternovska 2005). In this task, a *given structure*, which might have an empty vocabulary, is expanded with interpretations of new vocabulary symbols to satisfy a specification. Complexity-wise, MX lies in-between model checking (full structure is given) and satisfiability (no structure is given). The task generalizes to the formalism of Modular Systems.

Model Expansion (MX) Task Given: $\mathcal{B}|_\sigma$ and algebraic expression α with input symbols σ. Find: $\mathcal{B}$ such that $\mathcal{B}$ satisfies α. Structure $\mathcal{B}$ expands structure $\mathcal{B}|_\sigma$ and is called a *solution* of modular system α for a particular input $\mathcal{B}|_\sigma$.
Thus, the algebra with information flow may be called "a logic of hybrid MX tasks", and it will be interpreted over transition systems.[7]

[7] An interesting version is where only some inputs and outputs are specified, and some modules, even though *are known to be binary*, do not have input-output assignments. In this case, we obtain a collection of transition systems, one for each possible assignment, and both sceptical (under all input-output assignments) and brave reasoning (under some input-output assignment) can be studied.

Syntax Fix a relational vocabulary τ and a vocabulary of atomic module symbols τ_M so that $\tau \cap \tau_M = \varnothing$.
Let τ_P, where $\tau_P \subseteq \tau_M$, be a vocabulary of atomic module symbols where inputs are *not* specified. We call them *propositions*. Let τ_{act}, where $\tau_{act} \subseteq \tau_M$, be a vocabulary of atomic module symbols $M_i(\underline{X_{i_1}}, \ldots, X_{i_k})$, where inputs are underlined. We call them *actions*. For one module symbol M_i, we can potentially have both a proposition and several actions, depending on the choice of the inputs.
We define a calculus of binary relations as follows.

$$\alpha ::= \bot | M_i? | M_a | Z_j | \alpha + \alpha | \alpha \circ \alpha | \pi_\delta \alpha | \sigma_\Theta \alpha | \sim \alpha | \mu Z_j.\alpha \quad (4)$$

Here, M_i are propositions, M_a are actions, $\sim$ is a unary operation which is a special kind of negation, as is used in modal temporal logics. Variables Z_j range over actions. We require that Z_j occurs positively (under an even number of negations $\sim$) in $\mu Z_j.\alpha$. Requirements on $\pi_\delta \alpha$ and $\sigma_\Theta \alpha$ are as before. The calculus is essentially BSFP (Abu Zaid, Grädel, and Jaax 2014), but with selection and projection added, and where we use *modules* for both unary and binary relations, which makes the semantics much more complicated.

Semantics Define a transition system $\mathcal{T} := (V; (M_a^{\mathcal{T}})_a, (M?_i^{\mathcal{T}})_i)$ (parameterized by valuation $(\mathcal{V}, v)$ defined above) that has domain V which is the set of all τ-structures over a fixed domain, and it interprets all actions M_a as subsets of $V \times V$ denoted by $\llbracket M_a \rrbracket^{\mathcal{T},\mathcal{V},v}$, and all monadic propositions $M_i?$ by structures (now nodes in the transition graph) $\llbracket M_i? \rrbracket^{\mathcal{T},\mathcal{V},v} \subseteq V$ in which they are true. Module variables Z_j that occur free in α are interpreted as actions, i.e., as subsets of $V \times V$. Their interpretations are denoted $\llbracket Z_j \rrbracket^{\mathcal{T},\mathcal{V},v}$. We require that for each $M?_i$ (respectively, M_a), symbols in $vocab(M?_i)$ (respectively, $vocab(M_a)$, $vocab(Z_j)$) are interpreted by $(\mathcal{T}, \mathcal{V}, v)$ in the same way for all structures (= states of $\mathcal{T}$) $\mathcal{B}$ (respectively, pairs of structures) on which they are true. As before, we require that for any two τ-structures $\mathcal{A}_1$, $\mathcal{A}_2$ which coincide on $vocab(M_i)$, we have $\mathcal{A}_1 \in \mathcal{V}(v, M_i)$ iff $\mathcal{A}_2 \in \mathcal{V}(v, M_i)$. We define the extension $\llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}$ of formula α in $\mathcal{T}$ under valuation $(\mathcal{V}, v)$ inductively below. One can understand this definition as follows. First, v maps relational variables to symbols of the vocabulary τ, so that we can talk about τ-structures (that compose concrete modules). Second, $\mathcal{V}$ provides an interpretation to each atomic module, which is a set of structures as before (i.e., a concrete module). From now on, we view modules as either actions or propositions, depending on whether or not inputs-outputs are specified. Actions are the transitions in the transition system $\mathcal{T}$, and propositions are labels of the states of $\mathcal{T}$. Sequential composition $\circ$ and non-deterministic choice $+$ act as expected, projection adds non-determinism, similarly to $+$, and selection restricts the action to that where the interpretations of L_1 and L_2 are equal (there are three cases, when both L_1 and L_2 are inputs, both are outputs, and one is input, one is output), with interpretations as we would expect. The semantics of $\mu Z_j.\alpha$ is exactly like that of the least fixed point operator in the modal mu-calculus $L\mu$. An interesting operation is negation $\sim$. It acts like negation in modal logic, and its meaning will be more clear when we discuss the "two-sorted" version of (4). The semantics of atomic actions will be more clear when we explain the connections to the situation calculus.

$\llbracket \bot \rrbracket^{\mathcal{T},\mathcal{V},v} := \varnothing.$
$\llbracket M_i? \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{B}, \mathcal{B}) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid \mathcal{B} \in \mathcal{V}(v, M_i)$ for some $v\}$.
$\llbracket M_a \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{B}_1, \mathcal{B}_2) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid \mathcal{B}_1|_{\tau \setminus \varepsilon_{M_a}} = \mathcal{B}_2|_{\tau \setminus \varepsilon_{M_a}}$
and $\mathcal{B}_2 \in \mathcal{V}(v, M_a)$ for some $v\}$.
$\llbracket \alpha_1 + \alpha_2 \rrbracket^{\mathcal{T},\mathcal{V},v} := \llbracket \alpha_1 \rrbracket^{\mathcal{T},\mathcal{V},v} \cup \llbracket \alpha_2 \rrbracket^{\mathcal{T},\mathcal{V},v}.$
$\llbracket \alpha_1 \circ \alpha_2 \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{A}, \mathcal{B}) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid$
$\exists \mathcal{C}((\mathcal{A}, \mathcal{C}) \in \llbracket \alpha_1 \rrbracket^{\mathcal{T},\mathcal{V},v}$ and $(\mathcal{C}, \mathcal{B}) \in \llbracket \alpha_2 \rrbracket^{\mathcal{T},\mathcal{V},v})\}$.
$\llbracket \pi_\delta(\alpha) \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{B}_1, \mathcal{B}_2) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid$
$\exists \mathcal{C}_1 \exists \mathcal{C}_2 \; ((\mathcal{C}_1, \mathcal{C}_2) \in \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}, \mathcal{C}_1|_\delta = \mathcal{B}_1|_\delta$ and $\mathcal{C}_2|_\delta = \mathcal{B}_2|_\delta)\}$.
$\llbracket \sim \alpha \rrbracket^{\mathcal{T},\mathcal{V},v} = \{(\mathcal{B}, \mathcal{B}) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid \forall \mathcal{B}' \; (\mathcal{B}, \mathcal{B}') \notin \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}\}$,
no outgoing α-transition.
$\llbracket \mu Z_j.\alpha \rrbracket^{\mathcal{T},\mathcal{V},v} := \bigcap \big\{R \subseteq V^{\mathcal{T}} \times V^{\mathcal{T}} \; : \; \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V}[Z:=R],v} \subseteq R\big\}.$

$\llbracket \sigma_{L_1 \equiv L_2}(\alpha) \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{B}_1, \mathcal{B}_2) \in V^{\mathcal{T}} \times V^{\mathcal{T}} \mid 3$ cases:
1) $(\mathcal{B}_1, \mathcal{B}_2) \in \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}$ and $\{L_1, L_2\} \subseteq \sigma_\alpha$ and $\mathcal{B}_1 \models_{\mathrm{FO}} L_1 \equiv L_2$
2) $(\mathcal{B}_1, \mathcal{B}_2) \in \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}$ and $\{L_1, L_2\}) \subseteq \varepsilon_\alpha$ and $\mathcal{B}_2 \models_{\mathrm{FO}} L_1 \equiv L_2$
3) $L_1 \in \sigma_\alpha$ and $L_2 \in \varepsilon_\alpha$ and
$\exists \mathcal{C}((\mathcal{C}, \mathcal{B}_2) \in \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}, \; \mathcal{B}_1|_{\tau \setminus \{L_1\}} = \mathcal{C}|_{\tau \setminus \{L_2\}}, \mathcal{B}_2 \models_{\mathrm{FO}} (L_1 \equiv L_2)).$

Case 3 expresses Feedback from output L_2 to input L_1.

Example 5. To illustrate transitions using our examples, in (3), first $M_{\mathrm{HC}}(\underline{V}, \underline{X}, Y)$ makes transition by producing possibly several Hamiltonian Circuits. The interpretation of the output vocabulary, $\{Y\}$ changes, everything else is transferred by inertia. Then each resulting structure is taken as in input to 2-Colouring, $M_{2\mathrm{Col}}(\underline{V}, Y, Z, T)$, where edges in the cycle, Y, are "fed" to the second argument of $M_{2\mathrm{Col}}$, although this is hidden from the outside observer by the existential quantifier in (3). The second module produces non-deterministic transitions, one for each generated colouring.

Following (Abu Zaid, Grädel, and Jaax 2014), we define:

$$D :=\sim \bot, \quad \pi_1(\alpha) \; := \sim\sim \alpha.$$

By these definitions,

$$\llbracket D \rrbracket^{\mathcal{T},\mathcal{V},v} = \{(\mathcal{B}, \mathcal{B}) \in V^{\mathcal{T}} \times V^{\mathcal{T}}\},$$
$$\llbracket \pi_1(\alpha) \rrbracket^{\mathcal{T},\mathcal{V},v} := \{(\mathcal{B}, \mathcal{B}) \in V^{\mathcal{T}} \times V^{\mathcal{T}} | \exists \mathcal{B}' \; (\mathcal{B}, \mathcal{B}') \in \llbracket \alpha \rrbracket^{\mathcal{T},\mathcal{V},v}\}.$$

That is, D is the diagonal and π_1 abbreviates projection onto the first argument of the binary relation (onto all the inputs). The latter operation identifies the states in V where there is an outgoing α-transition.

Two-Sorted Syntax The grammar (4) for the algebra with information flow can be equivalently represented in a "two-sorted" syntax, where expressions for state formulas ϕ and processes α are defined by mutual recursion.

$$\alpha ::= D \mid \varnothing \mid M_a \mid Z_j \mid \alpha + \alpha \mid \alpha \circ \alpha \mid \pi_\delta(\alpha) \mid \sigma_\Theta(\alpha) \mid \phi? \mid \mu Z_j.\alpha$$
$$\phi ::= M_i \mid X_i \mid \; \phi \vee \phi \mid \neg \phi \mid \langle \alpha \rangle \phi \mid \mu X_j.\phi \qquad (5)$$

Thus, we can write $\langle \alpha \rangle \phi$ (respectively, $[\alpha]\phi$) to express that after some (respectively, all) executions of modular system α, property ϕ holds. Notice that we have binary (for processes) and unary (for state formulas) fixed points.

The two representations of the algebra (one-sorted and two-sorted) are equivalent. It follows from a theorem from (Abu Zaid, Grädel, and Jaax 2014) that holds also in our setting, where we have modules for actions and propositions.

Theorem 1. *For every state formula ϕ in two-sorted syntax (5) there is a formula $\hat{\phi}$ in the minimal syntax (4) such that $\mathcal{T}, \mathcal{V}, v \models \phi$ iff $\mathcal{T}, \mathcal{V}, (v, v) \models \pi_1 \hat{\phi}$, and for every action formula α there is an equivalent formula $\hat{\alpha}$ in the minimal syntax.*

It also follows from (Abu Zaid, Grädel, and Jaax 2014) that the well-known Propositional Dynamic Logic (PDL) is a fragment of the logic introduced above.

Proposition 2. *The Propositional Dynamic Logic (PDL)*

$$\begin{array}{l} \alpha ::= M_a \mid \alpha + \alpha \mid \alpha \circ \alpha \mid \ \phi? \mid \alpha^* \\ \phi ::= M_i \mid \ \phi \vee \phi \mid \neg \phi \mid \langle \alpha \rangle \phi \end{array} \tag{6}$$

is a fragment of (5).

Proof. It is sufficient to express α^* since the other operations of PDL are a subset of (5). We have $\alpha^* := \mu Z.(D + Z \circ \alpha)$. □

Model Expansion and Model Checking Tasks for Modular Systems A very naive method to solve model expansion for a modular system α would be to guess a structure expanding the input, and to check if it satisfies the algebraic expression. However, one can also develop an algorithm that identifies the set of all states $S \subseteq V$ in the transition system where an algebraic expression holds. Such an algorithm can be developed for a fragment of the calculus that corresponds to the mu-calculus $L\mu$. The states in S will contain all expansions, for all instances. Then one can check whether a particular instance structure is in that set. A basic way to obtain S is by labelling the states of $\mathcal{T}$ by sub-expressions of α that hold in those states, going bottom-up on the structure of α. A better way is to use Binary Decision Diagrams (BDDs) and perform this labelling symbolically, as is standard in symbolic model checking.

Specification of Solving Quantified Boolean Formulas (QBFs) as an Algebraic Expression

In this section, we consider quantified boolean formulas, such as, for example, $\forall x(\exists y(x \times y) + \forall z(\neg x + z))$. We will demonstrate an algebraic expression with a fixed point that encodes an algorithm for evaluation of such QBFs. We also connect such an eveluation with symbolic model checking of a mu-calculus formula.

Example 6. We assume that the QBF formulas that are used in the input are well-formed, all negations are pushed inwards to appear in front of boolean atoms only. The "smallest" formulas appearing in the parse tree are propositional formulas that occur just after inner-most quantifier. We also assume that all variables are bound by either $\exists$ or $\forall$.

Modules used: parse tree of a QBF formula: $M_{\mathrm{PT}}(S, X_\phi, X_t, X_{t'}, X_{t''}, L_t, Q_t)$ and SAT $M_{\mathrm{SAT}}(X_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})$. Intuitively, unary relation S specifies a set of symbols, those used in QBFs and a special symbol ␣ , e.g. $\{x, y, z, \ldots, (,), \exists, \forall, \textvisiblespace\}$, Recall that QBF variables are elements of the domain. In each structure, all variables appearing in the QBF are partitioned into the interpretations of T_{out}, F_{out}, which stand for *true* and *false*, respectively. These two relations describe all possible satisfying assignments of each sub-formula.

$M_{\mathrm{SAT}}(X_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})$ generates all possible truth assignments of the sub-formula encoded by X_t (the so-called 'generate' part of the overall algebraic expression incoding solving QBFs), and each application of the selection operator in the recursive computation limits those assignments (the so-called "constraint" or "test" part).

The intepretations of T_{in} and F_{in} each contain exactly one element in each structure, and are constructed as follows. For all variables, for all structures in the module M_{SAT}, if the interpretaion of T_{out} contains that variable, then it appears in the interpretaion of T_{in} in that structure, and the interpretation of F_{in} is empty. Symmetrically, for all variables, for all structures in the module M_{SAT}, if the interpretaion of F_{out} contains that variable, then it appears in the interpretaion of F_{in} in that structure, and the interpretation of T_{in} is empty.

The interpretations of T_{in}, F_{in}, and those of T_{out}, F_{out} in each structure of M_{SAT} are constructed so that the truth value of a particular QBF variable is not true and false in the same structure, as it should be in a truth assignment.

The problem is represented (in the one-sorted syntax) as

$$\begin{array}{l} \exists X_t \exists X_{t'} \exists X_{t''} \exists L_t \exists Q_t \exists T_{\mathrm{in}} \exists F_{\mathrm{in}} [\\ M_{\mathrm{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t) \wedge \mu Z.\Psi(Z, M_{\mathrm{PT}}, M_{\mathrm{SAT}})], \end{array} \tag{7}$$

where Z is a module variable of the form $Z(X_t, Q_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})$ over which recursive iteration is performed. Existentially quantified variables are not visible "from the outside". Thus, we have S and X_ϕ on the input, and T_{out}, F_{out} on the output.

Expression $\mu Z.\Psi(Z, M_{\mathrm{PT}}, M_{\mathrm{SAT}})$ specifies an algorithm for evaluating QBFs by recursion over their parse trees, with $\Psi(Z, M_{\mathrm{PT}}, M_{\mathrm{SAT}}) := \phi_l \vee \phi_\exists \vee \phi_\forall \vee \phi_+ \vee \phi_\times$.

Base case, leaf:

$$\begin{array}{l} \phi_l := \sigma_{(L_t \equiv `\langle \textvisiblespace \rangle' \wedge Q_t \equiv `\langle \textvisiblespace \rangle' \wedge X_{t'} \equiv \bot \wedge X_{t''} \equiv \bot)} [\\ M_{\mathrm{SAT}}(X_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}}) \\ \wedge M_{\mathrm{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t)]. \end{array}$$

We use $\bot$ for predicate constant "false". The selection above requires that the label of the leaf is ␣ , there are no quantified variables ($Q_t \equiv `\langle \textvisiblespace \rangle'$) there are no child sub-formulas ($X_{t'} \equiv \bot \wedge X_{t''} \equiv \bot$).

Module M_{SAT} solves propositional satisfiability of the sub-formulas X_t, where the parse tree is given by M_{PT}.

Case $\forall$:

$$\begin{array}{l} \phi_\forall := \sigma_{(L_t \equiv `\langle \forall \rangle' \wedge X_{t''} \equiv \bot)} [M_{\mathrm{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t) \\ \wedge \sigma_{Q_t \equiv T_{in}} [Z(X_{t'}, Q_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})] \\ \wedge \sigma_{Q_t \equiv F_{in}} [Z(X_{t'}, Q_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})]]. \end{array}$$

Case $\exists$ is very similar, except disjunction is used instead of conjunction:

$$\begin{array}{l} \phi_\exists := \sigma_{(L_t \equiv `\langle \exists \rangle' \wedge X_{t''} \equiv \bot)} [M_{\mathrm{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t) \\ \wedge [\sigma_{Q_t \equiv T_{in}} [Z(X_{t'}, Q_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})] \\ \vee \sigma_{Q_t \equiv F_{in}} [Z(X_{t'}, Q_t, T_{\mathrm{in}}, F_{\mathrm{in}}, T_{\mathrm{out}}, F_{\mathrm{out}})]]]. \end{array}$$

Conjunction case ($\times$):

$$\begin{aligned}&\phi_\times := \sigma_{(L_t \equiv `\langle\times\rangle')}[M_{\text{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t)\\ &\wedge Z(X_{t'}, Q_t, T_{\text{in}}, F_{\text{in}}, T_{\text{out}}, \overline{F_{\text{out}}})\\ &\wedge Z(X_{t''}, Q_t, T_{\text{in}}, F_{\text{in}}, T_{\text{out}}, F_{\text{out}})].\end{aligned}$$

Disjunction case ($+$):

$$\begin{aligned}&\phi_+ := \sigma_{(L_t \equiv `\langle+\rangle')}[M_{\text{PT}}(\underline{S}, \underline{X_\phi}, X_t, X_{t'}, X_{t''}, L_t, Q_t)\\ &\wedge[Z(X_{t'}, Q_t, T_{\text{in}}, F_{\text{in}}, T_{\text{out}}, \overline{F_{\text{out}}})\\ &\wedge Z(X_{t''}, Q_t, T_{\text{in}}, F_{\text{in}}, T_{\text{out}}, F_{\text{out}})]].\end{aligned}$$

Temporal Model Checking to Solve QBFs

Since, in the dynamic version, we interpret algebrac expressions over transition systems, and the specification above (7) is a modal mu-calculus $L\mu$ formula, we can use temporal logic model checking to solve QBFs. Each state in the transition system is a structure over the entire vocabulary. The main difference is that instead of simple propositions that hold in states of the transition system, we have potentially computationally complex modules that have to be checked in those states. For examle, in the case of our QBF example, we need to check whether leaf formulas hold in the states, that is we need to use the propositional satisfiability module M_{SAT}.

If we want to use symbolic model checking, we need to represent subsets of the set of all states, i.e., subsets of the set of all structures as BDDs. This is possible since each structure (a state in the stransition system) can be viewed as a set of ground atomic formulas. The transition relation in the QBF example is genarated dynamically, through the binary module variable Z, and it follows the subformula relation in the parse tree of the QBF on the input. A BDD for this relation can also be constructed, and the rest is standard.

Connection to the Situation Calculus and GOLOG

The Situation Calculus(see (Reiter 2001)) is a second order language that gives an axiomatic way to describe transition systems. All changes are the result of *actions*. A possible world history, which is a sequence of actions, is represented by a first order term called a *situation*. The constant S_0 is used to denote the *initial situation*. A binary function symbol $do(\alpha, s)$ denotes the successor situation to s resulting from performing the action α. Predicate symbol $Poss(a, s)$ specifies conditions on actions being executable is a situation. Relations (functions) whose truth values vary from situation to situation are called *fluents*. They are denoted by predicate (function) symbols taking a situation term as their last argument. Axiomatizations of a dynamic domain include: (1) *Action precondition axioms*, one for each primitive action. These characterize the relation $Poss$, and give the preconditions for the performance of an action in a situation. (2) *Successor state axioms*, one for each fluent. These capture the causal laws of the domain, together with a solution to the frame problem. (3) *Unique names axioms for the primitive actions*, stating that different names for actions denote different actions. (4) Axioms describing the *initial situation*. All the axiom together specify a transition system.

Connection to the Situation Calculus Atomic module actions in 4 are very similar to the actions of the situation calculus. Module M_a "looks" at the interpretations of the input symbols σ_{M_a} in the structure $\mathcal{B}_1$ (current "situation"), and expands $\mathcal{B}_1|_{\sigma_{M_a}}$ to produce interpretation of ε_{M_a} "recorded" in the structure $\mathcal{B}_2$ (one of the "M_a-successor" "situations"). Interpretations of all other symbols, including those in σ_{M_a}, stay the same, and get transferred from $\mathcal{B}_1$ to $\mathcal{B}_2$ *by inertia*. This is similar to the frame axioms in the situation calculus as described in R. Reiter's book (Reiter 2001). Notice that, similarly to the situation calculus, inertia is applied to *atomic* modules only. Each structure in module M_a may be seen as composed from its σ_{M_a} part "checked" in the structure $\mathcal{B}_1$, and its ε_{M_a} part "recorded" in the structure $\mathcal{B}_2$. Notice also that because of inertia, successor structure $\mathcal{B}_2$ contains information about both σ_{M_a} and ε_{M_a} part of a structure in M_a. Thus, the union of all M_a-successor structures, when limited to $vocab(M_a)$, is the entire module M_a.

GOLOG (Levesque et al. 1997) is a situation calculus-based language for defining complex actions using user-specified primitive actions. It can be described as:

$$\delta ::= a \mid \delta;\delta \mid \ \delta;\delta \ \mid \delta^* \mid \pi x.\delta \mid \phi? \mid Z(\bar{t}) \mid \text{Proc } Z(\bar{t})\delta, \qquad (8)$$

where a is a primitive action, $\delta;\delta$ is a sequence, $\delta|\delta$ is choice, δ^* non-deterministic iteration, $\pi x.\delta$ is nondeterministic choice of argument, $\phi?$ is test action, $Z(\bar{t})$ is procedure call, and Proc $Z(\bar{t})\delta$ is procedure description (that includes recursion). Constructs **if** ϕ **then** δ_1 **else** δ_2 and **while** ϕ **do** δ are definable through the other operations. The semantics of procedures is given through a least fixed point construct, which is like the binary fixed point construct $\mu Z.\alpha$. Since iteration δ^* is definable, (8) is a subset of the operations in the first line of (5). Another interesting observation is that unary fixed point properties, as in the second line of (5), were already formulated, in the context of the situation calculus, in (De Giacomo, Ternovskaia, and Reiter 1997).

Related Work

Literature on modularity and combined problem solving is enormous, and we do not attempt to review it here. Instead, we discuss only the most relevant work.

A large part of this paper studies Codd's algebra in the context of Model Expansion task. These tasks are common in AI planning, scheduling, logistics, supply chain management, etc. Java programs, if they are of input-output type, can be viewed as model expansion tasks, regardless of what they do internally. ASP systems, e.g., Clasp (Gebser, Kaufmann, and Schaub 2012) mostly solve model expansion, and so do CP languages such as Essence (Frisch et al. 2008), as shown in (Mitchell and Ternovska 2008). Problems solved in ASP competitions are mostly in model expansion form. CSP in the traditional AI form (respectively, in the homomorphism form) is representable by model expansion where mappings to domain elements (respectively, homomorphism functions) are expansion functions.

The notion of a module in (Tasharrofi and Ternovska 2011) is mathematically the same as in the current paper. A module there is a class of structures. Information propagation there happens through equivalent vocabulary symbols. However, the set of operations in that paper is smaller than in our paper here (there is no recursion, only a particular kind of selection (Feedback) is used). Moreover, connections to Codd's relational algebra or modal temporal logic are not established there.

The paper that originally inspired our Modular Systems framework is (Järvisalo et al. 2009), but we developed a model-theoretic approach and provided additional operations (in (Järvisalo et al. 2009), projections and sequential compositions are possible, but not the other operations used here (unions, selections, recursion,etc.). Compositions in (Lierler and Truszczyński 2015) are products only.

Unlike, e.g. dlvhex programs (Eiter et al. 2006), recursion in our algebra is over a module variable Z_i, not a predicate variable. Thus, the purely syntactic requirement of Z_i to occur positively in E is sufficient to ensure monotonicity.

Multi-Context Systems (MCSs) (Brewka and Eiter 2007) combine knowledge bases in arbitrary languages, under arbitrary semantics (that do not have to be model-theoretic) through rules of logic programming with negation as failure, which is a totally different phylosophy of combinations.

Conclusion

By adding information flow, we uncovered complex choreographies in the stillness of the simple declarative language of Modular Systems. There are significant benefits in studying modal fragments of our algebra (i.e., algebra with information flow). Such fragments often possess good model-theoretic and algorithmic properties. The main practical implication of the duality is that traditional "dynamic" techniques such as situation calculus, automata theory, temporal logic model checking etc. can be used to answer questions about the "static" formalism. Model checking for the dynamic system, for an important fragment of the logic, can be used to solve model expansion for the "still" version. Thus, the dynamic view opens up possibilities of new algorithms.

References

Abseher, M.; Bliem, B.; Charwat, G.; Dusberger, F.; Hecher, M.; and Woltran, S. 2014. The D-FLAT system for dynamic programming on tree decompositions. In Fermé, E., and Leite, J., eds., *Logics in Artificial Intelligence - 14th European Conference, JELIA 2014, Funchal, Madeira, Portugal, September 24-26, 2014. Proceedings*, volume 8761 of *Lecture Notes in Computer Science*, 558–572. Springer.

Abu Zaid, F.; Grädel, E.; and Jaax, S. 2014. Bisimulation safe fixed point logic. In Goré, R.; Kooi, B. P.; and Kurucz, A., eds., *Advances in Modal Logic 10, invited and contributed papers from the tenth conference on "Advances in Modal Logic," held in Groningen, The Netherlands, August 5-8, 2014*, 1–15. College Publications.

Brewka, G., and Eiter, T. 2007. Equilibria in heterogeneous nonmonotonic multi-context systems. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI'07) - Volume 1*, 385–390. AAAI Press.

Charwat, G., and Woltran, S. 2015. Efficient problem solving on tree decompositions using binary decision diagrams. In Francesco Calimeri, Giovambattista Ianni, M. T., ed., *Logic Programming and Nonmonotonic Reasoning, 13th International Conference, LPNMR 2015, Lexington, September 27-30, 2015. Proceedings*, Lecture Notes in Computer Science. Springer.

Dawar, A., and Gurevich, Y. 2002. Fixed point logics. *Bulletin of Symbolic Logic* 8(1):65–88.

De Giacomo, G.; Ternovskaia, E.; and Reiter, R. 1997. Nonterminating processes in the situation calculus. In *Proc. of the AAAI97 Workshop on Robots, Softbots, Immobots: Theories of Action, Planning and Control*, 18–28.

Eiter, T.; Ianni, G.; Schindlauer, R.; and Tompits, H. 2006. dlvhex: A prover for semantic-web reasoning under the answer-set semantics. In *2006 IEEE / WIC / ACM International Conference on Web Intelligence (WI 2006), 18-22 December 2006, Hong Kong, China*, 1073–1074. IEEE Computer Society.

Enderton, H. B. 1972. *A mathematical introduction to logic*. Academic Press.

Frisch, A. M.; Harvey, W.; Jefferson, C.; Martínez-Hernández, B.; and Miguel, I. 2008. Essence: A constraint language for specifying combinatorial problems. *Constraints* 13:268–306.

Gebser, M.; Kaufmann, B.; and Schaub, T. 2012. Conflict-driven answer set solving: From theory to practice. *Artificial Intelligence* 187-188:52–89.

Grädel, E.; Kolaitis, P. G.; Libkin, L.; Marx, M.; Spencer, J.; Vardi, M.; Venema, Y.; and Weinstein, S. 2007. *Finite Model Theory and Applications*. Springer.

Järvisalo, M.; Oikarinen, E.; Janhunen, T.; and Niemelä, I. 2009. A module-based framework for multi-language constraint modeling. In *Proceedings of the 10th International Conference on Logic Programming and Non-monotonic Reasoning (LPNMR'09)*, volume 5753 of *Lecture Notes in Computer Science (LNCS)*, 155–168. Springer-Verlag.

Levesque, H.; Reiter, R.; Lespérance, Y.; Lin, F.; and Scherl, R. 1997. GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming* 31:59–84.

Libkin, L. 2004. *Elements of Finite Model Theory*. Springer Verlag.

Lierler, Y., and Truszczyński, M. 2015. An abstract view on modularity in knowledge representation. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*.

Mitchell, D. G., and Ternovska, E. 2005. A framework for representing and solving NP search problems. In *Proc. AAAI'05*, 430–435.

Mitchell, D. G., and Ternovska, E. 2008. Expressiveness and abstraction in ESSENCE. *Constraints* 13(2):343–384.

Reiter, R. 2001. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems.* MIT Press.

Robertson, N., and Seymour, P. D. 1984. Graph minors. III. planar tree-width. *J. Comb. Theory, Ser. B* 36(1):49–64.

Tasharrofi, S., and Ternovska, E. 2011. A semantic account for modularity in multi-language modelling of search problems. In *Proceedings of the 8th International Symposium on Frontiers of Combining Systems (FroCoS)*, 259–274.

AGM-Style Revision of Beliefs and Intentions from a Database Perspective (Preliminary Version)

Marc van Zee and Dragan Doder
Computer Science and Communications,
University of Luxembourg, University of Luxembourg

Abstract

We introduce a logic for temporal beliefs and intentions based on Shoham's database perspective. We separate strong beliefs from weak beliefs. Strong beliefs are independent from intentions, while weak beliefs are obtained by adding intentions to strong beliefs and everything that follows from that. We formalize coherence conditions on strong beliefs and intentions. We provide AGM-style postulates for the revision of strong beliefs and intentions. We show in a representation theorem that a revision operator satisfying our postulates can be represented by a pre-order on interpretations of the beliefs, together with a selection function for the intentions.

1 Introduction

Recently there has been an increase in articles studying the dynamics of intentions in logic [Ditmarsch *et al.*, 2011; Icard *et al.*, 2010; Lorini and Herzig, 2008; van der Hoek *et al.*, 2007; Lorini *et al.*, 2009; Grant *et al.*, 2010]. Most of those papers take as a starting point the logical frameworks derived from Cohen and Levesque [1991], which in turn formalize Bratman's [1987] planning theory of intention. In this paper, we take a different starting point, and study the revision of intentions from a *database perspective* [Shoham, 2009]. The database perspective consists of a planner, a belief database and an intention database. Shoham [2016] describes it as "(...) a generalization of the AGM scheme for belief revision, (...). In the AGM framework, the intelligent database is responsible for storing the planner's beliefs and ensuring their consistency. In the enriched framework, there are two databases, one for beliefs and one for intentions, which are responsible for maintaining not only their individual consistency but also their mutual consistency." (p.48) Shoham further developed these ideas with Jacob Banks, one of his PhD students, and behavioral economist Dan Ariely in the intelligent calendar application Timeful, which attracted over \$6.8 million in funding and was acquired by Google in 2015[1], who aim to integrate it into their Calendar applications. As Shoham [2016] says himself: "The point of the story is there is a direct link between the original journal paper and the ultimate success of the company." (p.47) Thus, it seems clear that his philosophical proposal has lead to some success on the practical side. In this paper, we investigate whether his proposal can lead to interesting theoretical insights as well. More specifically, the aim of this paper is to develop a suitable formal theory for the belief and intention database in the database perspective, and to study belief and intention revision for this theory. Following Shoham's proposal, our methodology is to generalize AGM revision [Alchourron *et al.*, 1985] for temporal beliefs and intentions in terms of a representation theorem.

In the area of intention revision and reconsideration, Grant *et al.* [2010] combine intention revision with AGM-like postulates. There have also been a number of contributions applying AGM-style revision to action logics [Shapiro *et al.*, 2011; Jin and Thielscher, 2004; Scherl, 2005; Scherl and Levesque, 2003; Bonanno, 2007]. However, these proposals only characterize revision using a set of postulates, without proving representation theorems. There are also approaches that focus on the semantical level by postulating revision on a Kripke model [Baral and Zhang, 2005]. Both Icard *et al.* [2010] and van Zee *et al.* [2015b] develop a logic based on Shoham's database perspective and prove representation theorems, but we will analyze the shortcomings of their proposals in the next section.

The three main technical results of this paper are as follows: First we review two recent formalisations based on Shoham's database perspective, namely the IPS framework [Icard *et al.*, 2010] and the PAL framework [van Zee *et al.*, 2015a], and we discuss the shortcomings of these logics. Secondly, we extend PAL in order to define a coherence condition on strong beliefs and intentions, and we separate strong, or intention-independent, beliefs from weak beliefs. Lastly, we characterize revision of beliefs and intentions through AGM-style postulates and we prove a representation theorem relating the postulates for revision to an ordering among interpretations and a selection function that accommodates new intentions while restoring coherence. The three next sections of the paper are in line with these three results.

[1] http://venturebeat.com/2015/05/04/google-acquires-scheduling-app-timeful-and-plans-to-integrate-it-into-google-apps/

2 Preliminaries: The Database Perspective

We review Shoham's database perspective, and we discuss the main limitations of two recent formalizations. For the first formalization (IPS), the separation between strong and weak beliefs is problematic, and the representation theorem only holds for a specific type of beliefs. For the second formalization (PAL), intention revision is missing.

2.1 Shoham's Database Perspective

Shoham's database perspective contains a planner (e.g., a STRIPS-like planner) that is itself engaged in some form of practical reasoning. In the course of planning, it may add actions to be taken at various times in the future to an intention database and add observations to a belief database. The intentions are *future-directed intentions* of the form (a,t), meaning that action a will be executed at time t.[2] The beliefs are also time-indexed, and are of the form p_t, meaning that p is true at time t. Shoham treats the planner as a "black box": It provides the databases with input but its internal workings are unknown. Shoham proposes informal revision procedures for beliefs and intentions based on the following coherence conditions:

1. If two intended actions immediately follow one another, the earlier cannot have postconditions that are inconsistent with the preconditions of the latter.
2. If you intend to take an action you cannot believe that its preconditions do not hold.
3. If you intend to take an action, you believe that its postconditions hold.

Note that requirement 2 and 3 describe an asymmetry between pre-and postconditions: The postconditions are believed to be true after an intended action, but the preconditions may not. Therefore, we might think of the requirements as one of "optimistic" beliefs. According to Shoham [2009]: "It is a good fit with how planners operate. Adopting an optimistic stance, they feel free to add intended actions so long as they are consistent with current beliefs." (p.7)

2.2 Icard *et al.* (IPS)

Icard *et al.* [2010] develop a "formal semantical model to capture action, belief and intention, based on the 'database perspective"' (p.1). They assume a set of atomic sentences $\mathsf{Prop} = \{p,q,r,\ldots\}$ and deterministic primitive actions $\mathsf{Act}=\{a,b,c,\ldots\}$. Entries in the belief database are represented by a language generated from:

$$\varphi := p_t \mid pre(a)_t \mid post(a)_t \mid Do(a)_t \mid \Box\varphi \mid \varphi\wedge\varphi \mid \neg\varphi$$

with $p \in \mathsf{Prop}, a \in \mathsf{Act}$, and $t \in \mathbb{Z}$. p_t means that p is true at time t, $Do(a)_t$ means that the agent does action a at time t, and $pre(a)_t$ and $post(a)_t$ represent respectively the precondition and postcondition of action a at time t.

Icard *et al.* use a semantics of *appropriate paths*. They define $P = \mathcal{P}(\mathsf{Prop} \cup \{pre(a), post(a) : a \in \mathsf{Act}\})$, and a *path* $\pi : \mathbb{Z} \to (P \times \mathsf{Act})$ as a mapping from a time point to a set of proposition-like formulas true at that time (denoted $\pi(t)_1$) and the next action a on the path (denoted $\pi(t)_2$). They define an equivalence relation $\pi \sim_t \pi'$, which means that π and π' represent the same situation up to t. Using this, they propose a notion of appropriateness:

Definition 1 (Appropriate Set of Paths)**.** *A set of paths* Π *is* appropriate *iff for all* $\pi \in \Pi$*:*

- *If* $\pi(t)_2 = a$*, then* $post(a) \in \pi(t+1)_1$*,*
- *If* $pre(a) \in \pi(t)_1$*, then there exists* $\pi' \sim_t \pi$ *s.t.* $\pi'(t)_2 = a$*.*

The truth definition $\models_\Pi$ is defined relative to an appropriate set of paths Π, and the modality is defined as follows:

$$\pi,t \models_\Pi \Box\varphi, \text{ iff for all } \pi' \in \Pi, \text{ if } \pi \sim_t \pi' \text{ then } \pi',t \models \varphi.$$

A model for a formula is an appropriate set of paths. They introduce an intention database $I = \{(a,t),\ldots\}$ as a set of action-time pairs (a,t) and put the following coherence condition on their logic:

$$Cohere^*(I) = \Diamond \bigwedge_{(a,t)\in I} pre(a)_t.$$

This captures the intuition that an agent considers it possible to carry out all intended actions. They state that a set of models is coherent if and only if there exists a model in which $Cohere^*(I)$ is true. IPS distinguishes *intention-contingent*, or weak, beliefs from *non-contingent*, or strong, beliefs. Contingent beliefs B^I are obtained from a belief-intention database (B,I) as follows: $B^I = Cl(B \cup \{Do(a)_t : (a,t) \in I\})$. In order to switch from belief bases to an appropriate set of paths, Icard *et al.* introduce the functions ρ and β: "Given a set of formulas B, we can consider the set of paths on which all formulas of B hold at time 0, denoted $\rho(B)$. Conversely, given a set of paths Π, we let $\beta(\Pi)$ be defined as the set of formulas valid at 0 in all paths in Π." (p.3)

The first issue with IPS is that the definition of non-contingent beliefs is problematic for coherence. The following example shows that non-contingent beliefs that are dependent on the actual path can lead to a coherent agent with inconsistent weak beliefs.

Example 1. *Consider an IPS belief-intention base* (B,I) *with the belief base* $B = \{p_1, \neg\Diamond(do(a)_2 \wedge p_1)\}$ *and intention base* $I = \{(a,2)\}$*. While* B *is consistent with* $\Diamond pre(a)_2$*,* B^I *is inconsistent since* $\neg\Diamond(do(a)_2 \wedge p_1) \wedge do(a)_2$ *derives* $\neg p_1$*, but this is inconsistent with the initial belief* p_1*.*

In Section 3 we will define our notion of coherence that together with a separation between strong and weak beliefs resolves this problem.

The second issue with IPS is that the definition of ρ is circular, and as a result it does not seem to be possible for all formulas of their logic. Consider the following example.

Example 2. *Suppose* $B = \{\Box p_1 \vee \Box\neg p_1\}$*. The set of paths* $\rho(B)$ *contains all paths* π *for which* $\pi \models B$*, i.e.* $\pi \models \Box p_1 \vee \Box\neg p_1$*. Take such a path* π *arbitrary, so either* $\pi \models \Box p_1$ *or* $\pi \models \Box\neg p_1$*. Suppose* $\pi \models \Box p_1$*. Then it follows that for all* $\pi' \sim_t \pi$ *we have* $\pi' \models p_1$*. On the other hand, if* $\pi \models \Box\neg p_1$*, then it follows that for all* $\pi' \sim_t \pi$ *we have* $\pi' \models \neg p_1$*. So in both cases we end up with two different sets for* $\rho(B)$*. In other words, the set of paths* $\rho(B)$ *is not defined.*

[2] The notion of intention here is clearly quite restrictive and important characteristics of intentions are missing. See the conclusion for a discussion.

Essentially, according to IPS, $\rho(B) = \{\pi \mid \pi \models_\Pi B\}$. The set of appropriate paths for a belief base B is thus constructed using $\models_\Pi$. However, $\models_\Pi$ is aleady defined relatively to some set of paths Π. Therefore, the definition of ρ is circular. It seems that the ρ function only works for belief bases containing no modalities (in other words, to construct a single path). We omit details for space constraints, but the construction of the canonical model in the proof of their representation theorem uses the ρ function to switch from a belief base to a set of paths (see Proof Sketch in the Appendix of Icard *et al.* [2010]). Therefore, the representation theorem does not hold for all formulas of the logic, since it is not possible to apply the function ρ to all beliefs.

Summarizing, we recognize two shortcoming of the IPS framework as a formal basis for the database perspective: The definition of contingent beliefs is problematic, and the representation theorem does not hold for belief bases containing modalities.

2.3 van Zee *et al.* (PAL)

van Zee *et al.* [2015b] develop Parameterized-time Action Logic (PAL) as an alternative to IPS, claiming that the IPS framework contains an unsound axiom and that their logic is noncompact [van Zee *et al.*, 2015a]. They define $\mathsf{Act} = \{a, b, c, \ldots\}$ as a finite set of deterministic primitive actions, and $\mathsf{Prop} = \{p, q, r, \ldots\} \cup \{pre(a), post(a) \mid a \in \mathsf{Act}\}$ as a finite set of propositions. They denote atomic propositions with χ. PAL differs syntactically from IPS in that the $\Box$-modality is indexed by a time-point. Their language $\mathscr{L}_{PAL}$ is inductively defined by the following BNF grammar:

$$\varphi ::= \chi_t \mid do(a)_t \mid \Box_t\varphi \mid \varphi \wedge \varphi \mid \neg\varphi$$

PAL uses a CTL*-like tree semantics consisting of a tree $T = (S, R, v, act)$ where S is a set of states, R is an accessibility relation that is serial, linearly ordered in the past and connected, $v : S \to 2^{\mathsf{Prop}}$ is a valuation function from states to sets of propositions, and $act : R \to \mathsf{Act}$ is a function from accessibility relations to actions, such that actions are deterministic, i.e. if $act((s, s')) = act((s, s''))$, then $s' = s''$. Similarly to Icard *et al.*, PAL uses an equivalence relation $\sim_t$ on paths. Using this equivalence relations they define a model as a pair (T, π) on which the same conditions hold as an IPS model. PAL formulas are evaluated in a model on a path. van Zee *et al.* axiomatize PAL and show that it is sound and strongly complete, i.e. $T \vdash \varphi$ iff $T \models \varphi$. They characterize AGM belief revision in this logic and restrict time up to some t. Using these constraints, they are able to represent a belief set B as a propositional formula ψ such that $B = \{\varphi \mid \psi \vdash \varphi\}$ and they prove the Katsuno and Mendelzon (KM) [1991] and the Darwiche and Pearl (DP) [Darwiche and Pearl, 1997] representation theorems.

van Zee *et al.* only consider revision of PAL formulas and thus do not consider the problem of intention revision, and therefore also does not distinguish between strong and weak beliefs. However, PAL uses a standard branching time (CTL*-like [Reynolds, 2002]) semantics, and it therefore also does not suffer from the shortcomings that we addressed in the previous subsection. Therefore, we take PAL as a starting point and use the remainder of this paper to extend it so that we can study intention revision.

3 The Belief-Intention Database

We assume an intention database $I = \{(a, t), \ldots\}$ consisting of time-indexed actions. We next define a coherence condition on beliefs and intentions in PAL, and we separate strong and weak beliefs.

3.1 The Coherence Condition

We first demonstrate that $Cohere^*(I)$ of IPS (Section 2.2) is too permissive because it allows models in which intentions are not jointly executable.

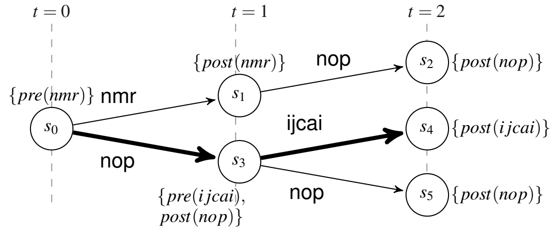


Figure 1: Example Model from $t = 0$ to $t = 2$.

Example 3 (Running Example). *An agent is considering to attend the NMR workshop at time 0 and the IJCAI conference at time 1. Although it would like to attend both events, there is insufficient budget available. Consider a possible partial semantic model of this situation depicted in Figure 1, where the thick path represents the actual path. In the actual path, the agent believes it does nothing at time 0 and attends IJCAI at time 1. It also considers it possible to attend the NMR workshop at time 0 and do nothing at time 1 in an alternative path. However, it does not consider it to be possible to attend both events.*

Suppose the agent of the running example has two intentions: $I = \{(nmr, 0), (ijcai, 1)\}$. Intuitively, the agent's intentions do not cohere with its beliefs, because it believes it cannot execute them both due to insufficient budget. However, according to $Cohere^*(I)$ the agent is coherent because the preconditions of all intentions hold on some path (namely the current path). Thus, $Cohere^*(I)$ does not fulfill Shoham's coherence condition 1 (Section 2.1). More specifically, the problem is that it is not possible to define the precondition of a set of actions in terms of preconditions of individual actions, because it cannot be ensured that all the intentions are fulfilled on the same path as well. Therefore, in order to formalize a coherence condition in PAL, we extend the language with preconditions of finite action sequences, which ensures that after executing the first action, the precondition for the remaining actions are still true. We modify the language, the definition of a model, the axiomatization, and we show that the new axiomatization is sound and strongly complete. We call the new logic PAL-P (Parameterized-time Action Logic with extended Preconditions).

Definition 2 (PAL-P Language). *The language $\mathscr{L}$ is obtained from $\mathscr{L}_{PAL}$ by adding $\{pre(a, b, \ldots)_t \mid \{a, b, \ldots\} \subseteq \mathsf{Act}, t \in \mathbb{N}\}$ to the set of propositions. Moreover, $Past(t)$ consists of boolean combinations of $p_{t'}, pre(a, b, \ldots)_{t'}, \Box_{t'}\varphi$ and $do(a)_{t'-1}$ where $t' \leq t$ and φ is some formula from $\mathscr{L}$.*

We also extend the definition of a model accordingly.

Definition 3 (PAL-P Model)**.** *A* model *is a pair* (T, π) *with* $T = (S, R, v, act)$ *such that for all* $\pi \in T$ *the following holds:*

1. *If* $act(\pi_t) = a$, *then* $post(a) \in v(\pi_{t+1})$,
2. *If* $pre(a) \in v(\pi_t)$, *then there is some* π' *in* T *with* $\pi \sim_t \pi'$ *and* $act(\pi'_t) = a$,
3. *If* $pre(\ldots, a, b)_t \in v(\pi_t)$, *then* $pre(\ldots, a)_t \in v(\pi_t)$,
4. *If* $pre(a, b, \ldots)_t \in v(\pi_t)$, *then there is some* π' *in* T *with* $\pi \sim_t \pi'$, $act(\pi'_t) = a$, *and* $pre(b, \ldots)_{t+1} \in v(\pi'_{t+1})$.

We refer to models of PAL-P with $m_1, m_2, \ldots$, *we refer to sets of models with* $M_1, M_2, \ldots$, *and we refer to the set of all models with* $\mathbb{M}$.

Definition 4. *The logic PAL-P consists of the all the axiom schemas and rules of PAL [van Zee* et al.*, 2015b] (Def. 7), and the following two:*

$$pre(\ldots, a, b)_t \rightarrow pre(\ldots, a)_t \qquad (A11)$$

$$(pre(a, b, \ldots)_t \wedge do(a)_t) \rightarrow pre(b, \ldots)_{t+1} \qquad (A12)$$

The relation $\vdash$ *is defined in the usual way with the restriction that necessitation can be applied to theorems only.*

Theorem 1 (Completeness Theorem)**.** *The logic PAL-P is sound and strongly complete, i.e.* $T \vdash \varphi$ *iff* $T \models \varphi$.[3]

The proof of the theorem is a direct extension the proof of van Zee *et al.* with the new axioms and the new conditions on a model.

Note that it is not directly possible in PAL-P to express preconditions for actions that do not occur directly after each other. In order to do so, we simply make a disjunction over all possible action combinations in the time points in between the actions. Thus, if for instance $\mathsf{Act} = \{a, b\}$ and $I = \{(a, 1), (b, 3)\}$, then $Cohere(I) = \Diamond_0 \bigvee_{x \in \mathsf{Act}} pre(a, x, b)_1 = \Diamond_0(pre(a, a, b)_1 \vee pre(a, b, b)_1)$.[4]

Definition 5 (Coherence)**.** *Given an intention database* $I = \{(b_{t_1}, t_1), \ldots, (b_{t_n}, t_n)\}$ *with* $t_1 < \ldots < t_n$, *let*

$$Cohere(I) = \Diamond_0 \bigvee_{\substack{a_k \in Act : k \notin \{t_1, \ldots, t_n\} \\ a_k = b_k : k \in \{t_1, \ldots, t_n\}}} pre(a_{t_1}, a_{t_1+1}, \ldots, a_{t_n})_{t_1}.$$

For a given set of models M, *we say that* (M, I) *is* coherent *iff there exists some* $m \in M$ *with* $m \models Cohere(I)$.

We now show that the new coherence conditions correctly specifies that the agent of our running example is not coherent.

Example 4 (Continued)**.** *The models of the agent of our running example are not coherent with the intention database* $I = \{(nmr, 0), (ijcai, 1)\}$, *because the agent does not have the possibility to jointly execute both intentions (i.e. the preconditions for both actions together is false). Thus, none of the models satisfy* $\Diamond_0 pre(nmr, ijcai)_0$, *even though they satisfy* $\Diamond_0 pre(nmr)_0 \wedge \Diamond_0 pre(ijcai)_1$.

3.2 Strong and Weak Beliefs

The idea behind strong beliefs is that they represent the agent's ideas about what is inevitable, no matter how it would act in the world. Weak beliefs, on the other hand, represent the beliefs of an agent that are the consequence of its planned actions [Van der Hoek and Wooldridge, 2003]. Formally, we define strong beliefs at some time t as formulas that start either with $\Diamond_t$ or $\Box_t$. The set of all *strong beliefs* $\mathbb{B}_t$ in time t for $\mathscr{L}_{PAL}$ is inductively defined by the following BNF grammar:

$$\varphi ::= \Box_t \psi \mid \varphi \wedge \varphi \mid \neg \varphi,$$

where $\psi \in \mathscr{L}_{PAL}$ and $t \in \mathbb{N}$. A *strong belief set in t* is $B_t \subseteq \mathbb{B}_t$. In the remainder of this paper, we assume $t = 0$ and we simply write $\mathbb{B}$ and B to abbreviate $\mathbb{B}_0$ and B_0.

The weak beliefs $WB(B, I)$ are obtained from the strong beliefs by adding beliefs that are contingent on the intentions of the agent. In other words, the agent weakly believes everything that it strongly believed and moreover that all intentions will be realised, and everything that follows from this:

$$WB(B, I) = Cl(B \cup \{do(a)_t \mid (a, t) \in I\}).$$

Example 5 (Continued)**.** *Some example strong beliefs of the agent of Figure 1 are* $\Diamond_0 do(nmr)_0$, $\Diamond_0 do(ijcai)_1$, *and* $\Box_0 \Diamond_1 post(nop)_2$. *If the agent has the intention database* $I = \{(ijcai, 1)\}$, *then* $post(ijcai)_2$ *is one of its weak beliefs.*

4 Belief and Intention Revision

Postulates Following KM, we fix a way of representing a belief set B consisting of strong beliefs by a propositional formula ψ such that $B = \{\varphi \mid \psi \vdash \varphi\}$. Since intentions and beliefs that have been added by a planner are naturally bounded up to some time point t, we define a bounded revision function and we restrict the syntax and semantics of PAL-P up to a specific time point. As a consequence, it is then possible to obtain the single formula ψ for a set of strong beliefs B (Corollary 1). We first define some notation that we use in the rest of this paper.

Definition 6. *An agent is a pair* (ψ, I) *consisting of a belief formula* ψ, *and an intention base* I. $\mathbb{A}$ *denotes the set of all agents,* $\mathbb{B}$ *denotes the set of all strong beliefs,* $\mathbb{I}$ *denotes the set of all intentions, and* $\mathbb{D}$ *denotes the set of all intention databases. We denote* $\mathbb{A}, \mathbb{B}, \mathbb{I}$, *and* $\mathbb{D}$ *bounded up to t with respectively* $\mathbb{A}^{|t}, \mathbb{B}^{|t}, \mathbb{I}^{|t}$, *and* $\mathbb{D}^{|t}$. *However, if the restriction is clear from context, we may omit the superscript notation.*

We now define a bounded revision function $*_t$ revising an agent (ψ, I) with a tuple (φ, i) consisting of a strong belief φ and an intention i, denoted $(\psi, I) *_t (\varphi, i)$, where t is the maximal time point occurring in ψ, I, φ, and i.

[3] We provide the full proofs of all theorems and propositions in this paper in a separate technical report: http://www.dropbox.com/s/798g6ki0aqt3gds/ijcai2016proofs.pdf?dl=1

[4] Our construction of preconditions over action sequences may lead to a coherence condition involving a big disjunction. This is a drawback in terms of computational complexity. Alternatively, one may explicitly denote the time of each precondition, e.g. $pre(a, b)_{(t_1, t_2)}$. We chose the former since it is conceptually closer to the original syntax, but the latter can be implemented straightforwardly.

Definition 7 (Agent Revision Function)**.** *An* Agent revision function $*_t : \mathbb{A} \times (\mathbb{B} \times \mathbb{I}) \to \mathbb{A}$ *maps an agent, a strong belief formula, and an intention— all bounded up to t— to an agent bounded up to t such that if,*
$(\psi, I) *_t (\varphi, i) = (\psi', I')$,
$(\psi_2, I_2) *_t (\varphi_2, i_2) = (\psi'_2, I'_2)$,
then following postulates hold:
$(P1)$ ψ' *implies* φ.
$(P2)$ *If* $\psi \wedge \varphi$ *is satisfiable, then* $\psi' \equiv \psi \wedge \varphi$.
$(P3)$ *If* φ *is satisfiable, then* ψ' *is also satisfiable.*
$(P4)$ *If* $\psi \equiv \psi_2$ *and* $\varphi \equiv \varphi_2$ *then* $\psi' \equiv \psi'_2$.
$(P5)$ *If* $\psi \equiv \psi_2$ *and* $\varphi_2 \equiv \varphi \wedge \varphi'$ *then* $\psi' \wedge \varphi'$ *implies* ψ'_2.
$(P6)$ *If* $\psi \equiv \psi_2$*,* $\varphi_2 \equiv \varphi \wedge \varphi'$*, and* $\psi' \wedge \varphi'$ *is satisfiable, then* ψ'_2 *implies* $\psi' \wedge \varphi'$.
$(P7)$ (ψ', I') *is coherent.*
$(P8)$ *If* $(\psi', \{i\})$ *is coherent, then* $i \in I'$.
$(P9)$ *If* $(\psi', I \cup \{i\})$ *is coherent, then* $I \cup \{i\} \subseteq I'$.
$(P10)$ $I' \subseteq I \cup \{i\}$.
$(P11)$ *If* $I = I_2$*,* $i = i_2$*, and* $\psi' \equiv \psi'_2$*, then* $I' = I'_2$.
$(P12)$ *For all* I'' *with* $I' \subset I'' \subseteq I \cup \{i\}$*:*$(\psi', I'')$ *is not coherent.*

Postulates (P1)-(P6) are simply the KM postulates in our setting, which are equivalent to the AGM postulates [Katsuno and Mendelzon, 1991]. They also state that the revision of strong beliefs does not depend on the intentions. Postulates (P7)-(P10) also appear in IPS. Postulate (P7) states that the outcome of a revision should be coherent. Postulate (P8) states that the new intention i take precedence over all other current intentions; if possible, it should be added, even if all current intentions have to be discarded. Postulate (P9) and (P10) together state that if it is possible to simply add the intention, then this is the only change that is made. Postulate (P11) states that if we revise with the same i but with a different belief, and we end up with the same belief in both cases, then we also end up with the same intentions. Finally, (P12) states that we do not discard intentions unnecessarily. This last postulate is comparable to the *parsimony requirement* introduced by Grant *et al.* [2010].

Representation Theorem We next characterize all revision schemes satisfying (P1)-(P12) in terms of minimal change with respect to an ordering among interpretations and a selection function accommodating new intentions while restoring coherence. We bound models of strong beliefs up to t, which means that all the paths in the model are "cut off" at t. This ensures finitely many non-equivalent formulas for some belief set B. A *t-bounded model* $m^{|t} = (T^{|t}, \pi^{|t})$ is a model containing a tree T in which all paths, including π, have length t. Strong beliefs are about possibility and necessity, and they are independent of a specific path. Therefore, if a single path in a tree is a model of a strong belief, then all paths in this tree are models of this strong belief. Formally, a set of models of a strong belief M_{SB} satisfies the following condition:

$$\text{If } (T, \pi) \in M_{SB}, \text{ then } (T, \pi') \in M_{SB} \text{ for all } \pi' \in T.$$

A set of *t-bounded models of a strong belief* $M^{|t}_{SB}$ contains only t restricted models of a strong belief. We write $\mathbb{M}^{|t}_{SB}$ to denote the set of all sets of t-bounded models of strong beliefs. We now show that we can represent a set of models of strong beliefs by a single formula.

Lemma 1. *Let* $Ext(M^{|t}_{SB})$ *be the set of all possible extensions of a set of bounded model of strong beliefs* $M^{|t}_{SB}$ *to models, i.e.* $Ext(M^{|t}_{SB}) = \{m \in \mathbb{M} \mid m^{|t} \in M^{|t}_{SB}\}$*. Given a set of t-bounded models of strong beliefs* $M^{|t}_{SB}$*, there exists a strong belief formula* $form(M^{|t}_{SB})$ *such that* $Mod(form(M^{|t}_{SB})) = Ext(M^{|t}_{SB})$*.*

Corollary 1. *Given a t-bounded strong belief set B, there exists a formula* ψ *such that* $B = \{\varphi \mid \psi \vdash \varphi\}$*.*

Proof Sketch. For a given belief set B, we can show that there exists a set of t-bounded models of a strong belief $M^{|t}_{SB}$ s.t. $Ext(M^{|t}_{SB}) = Mod(B)$. If $\psi = form(M^{|t}_{SB})$, then $Mod(\psi) = Mod(B)$, and by the completeness theorem, $B = Cl(\psi)$. □

Given an intention database I, we define a selection function γ^t_I that tries to accommodate a new intention based on strong beliefs. The selection function specifies preferences on which intention an agent would like to keep in the presence of the new beliefs.

Definition 8 (Selection Function)**.** *Given an intention database I, a* selection function $\gamma^t_I : \mathbb{M}_{SB} \times \mathbb{I} \to \mathbb{D}$ *maps a set of models of a strong belief and an intention to an updated intention database—all bounded up to t— such that if* $\gamma^t_I(M^{|t}, \{i\}) = I'$*, then:*

1. $(M^{|t}, I')$ *is coherent.*
2. *If* $(M^{|t}, \{i\})$ *is coherent, then* $i \in I'$.
3. *If* $(M^{|t}, I \cup \{i\})$ *is coherent, then* $I \cup \{i\} \subseteq I'$.
4. $I' \subseteq I \cup \{i\}$.
5. *For all* I'' *with* $I' \subset I'' \subseteq I \cup \{i\}$*:*$(M^{|t}, I'')$ *is not coherent.*

The five conditions on the selection function are in direct correspondence with postulates (P7)-(P10), (P12) of the agent revision function $*_t$. Note that postulate (P11) doesn't have a corresponding condition in the definition above but is represented by the fact that the selection function takes the revised beliefs as input. That is, intention revision occurs after belief revision.

KM define a faithful assignment from a belief formula to a pre-order over models. Since we are also considering intentions, we extend this definition such that it also maps intentions databases to selection functions.

Definition 9 (Faithful assignment)**.** *A* faithful assignment *is a function that assigns to each strong belief formula* $\psi \in \mathbb{B}^{|t}$ *a total pre-order* $\leq^t_\psi$ *over* $\mathbb{M}$ *and to each intention database* $I \in \mathbb{D}^{|t}$ *a selection function* γ^t_I *and satisfies the following conditions:*

1. *If* $m_1, m_2 \in Mod(\psi)$*, then* $m_1 \leq^t_\psi m_2$ *and* $m_2 \leq^t_\psi m_1$.
2. *If* $m_1 \in Mod(\psi)$ *and* $m_2 \notin Mod(\psi)$*, then* $m_1 < m_2$.
3. *If* $\psi \equiv \phi$*, then* $\leq^t_\psi = \leq^t_\phi$.
4. *If* $T^{|t} = T^{|t}_2$*, then* $(T, \pi) \leq^t_\psi (T_2, \pi_2)$ *and* $(T_2, \pi_2) \leq^t_\psi (T, \pi)$.

Conditions 1 to 3 on the faithful assignment are the same as those of KM. Condition 4 ensures that we do not distinguish between models in the total pre-order $\leq_{\psi}^{t}$ whose trees are the same up to time t. This is essentially what is represented in the revision function by bounding the all input of the revision function $*_t$ up to t. Moreover, $\leq_{\psi}^{t}$ does not distinguish between models obtained by selecting two different paths from the same tree. This corresponds to the fact that we are using strong belief formulas in the revision, which do not distinguish between different paths in the same tree as well.

Theorem 2 (Representation Theorem). *An agent revision operator $*_t$ satisfies postulates (P1)-(P12) iff there exists a faithful assignment that maps each ψ to a total pre-order $\leq_{\psi}^{t}$ and each I to a selection function γ_I^t such that if $(\psi, I) *_t (\varphi, i) = (\psi', I')$, then:*

1. $Mod(\psi') = \min(Mod(\varphi), \leq_{\psi}^{t})$
2. $I' = \gamma_I^t(Mod(\psi'), i)$

Proof Sketch. We only sketch the proof of "$\Rightarrow$": Suppose that some agent revision operator $*_t$ satisfies postulates (P1)-(P12). Given models m_1 and m_2, let $(\psi, \emptyset) *_t (form(m_1^{|t}) \vee form(m_2^{|t}), \varepsilon) = (\psi', \emptyset)$. We define $\leq_{\psi}^{t}$ by $m_1 \leq_{\psi}^{t} m_2$ iff $m_1 \models \psi$ or $m_1 \models \psi'$. We also define γ_I^t by $\gamma_I^t(M^{|t}{}_{SB}, i) = I'$, where $(form(M^{|t}{}_{SB}), I) *_t (\top, i) = (\psi_2, I')$ (note that $\psi_2 \equiv form(M^{|t}{}_{SB})$).

Let us prove condition 4 of Definition 9. For $m_1 = (T, \pi)$ and $m_2 = (T_2, \pi_2)$, let ψ' be as above. Since $\psi, \psi' \in \mathbb{B}^{|t}$ and $T^{|t} = T_2^{|t}$, we have $m_1 \models \psi$ iff $m_2 \models \psi$ and $m_1 \models \psi'$ iff $m_2 \models \psi'$, so $m_1 \leq_{\psi}^{t} m_2$ and $m_2 \leq_{\psi}^{t} m_1$.

Following KM, one can show that conditions 1 to 3 from Definition 9 hold, and furthermore that $Mod(\psi') = \min(Mod(\varphi), \leq_{\psi}^{t})$. We now prove $I' = \gamma_I^t(Mod(\psi'), i)$. By our definition of γ_I^t we have that $(\psi', I) *_t (\top, i) = (\psi_2, \gamma_I^t(Mod(\psi'), i))$ (recall that $\psi' \equiv \psi_2$). Since $(\psi, I) *_t (\varphi, i) = (\psi', I')$, by (P11) we obtain that $I' = \gamma_I^t(Mod(\psi'), i)$. Using postulate (P7)-(P10) and (P12) we can prove that γ_I^t is a selection function. □

Finally, it turns out to be straightforward to formulate the DP postulates for iterated revision in our framework for the strong beliefs and to prove their representation theorem. Due to space constraints we have omitted the results, but they can be found in a separate technical report.[5]

5 Related Work

Grant *et al.* [2010] develop AGM-style postulates for belief, intention, and goal revision. They provide a detailed analysis and propose different reconsideration strategies, but restrict themselves to a syntactic analysis. Much effort in combining AGM revision with action logics (e.g., the Event Calculus [Mueller, 2010], Temporal Action Logics [Kvarnström, 2005], extensions to the Fluent Calculus [Thielscher, 2001], and extensions to the Situation Calculus (see [Patkos, 2010, Ch.2] for an overview)) concentrates on extending these action theories to incorporate *sensing* or *knowledge-producing actions*. Shapiro *et al.* [2011] extend the Situation Calculus to reason about beliefs rather than knowledge by introducing a modality B and shows that both the AGM postulates and the DP postulates are satisfied in this framework. A similar approach concerning the Fluent Calculus has been formalized by Jin and Thielscher [2004], and is further developed by Scherl [2005] and Scherl and Levesque [2003] by taking into account the frame problem as well. However, none of these approaches prove representations theorems linking revision to a total pre-order on models. Baral and Zhang [2005] model belief updates on the basis of semantics of modal logic S5 and show that their knowledge update operator satisfies all the KM postulates. Bonanno [2007] combines temporal logic with AGM belief revision by extending a temporal logic with a belief operator and an information operator. Both these approaches do not take action or time into account and do not prove representation theorems. The concept of strong beliefs has been discussed extensively in the literature, for instance in the story of *Little Nell* [McDermott, 1982] or a paradox found in *knowledge-based programs* [Fagin *et al.*, 1995] (see van der Hoek *et al.* [2003] for a detailed discussion).

6 Conclusion

We develop a logical theory for reasoning about temporal beliefs and intentions based on Shoham's database perspective. We propose postulates for revision of strong beliefs and intentions, and prove a representation theorem relating the postulates to our formal model. Icard *et al.* prove a comparable representation theorem, but we show in this paper that it does not hold in general. In their proof, they use a canonical model construction, while we proof our representation theorem using standard techniques from belief revision. It remains an open problem whether a canonical model construction is possible when proving the representation theorem of Icard *et al.*

For future work, we aim to extend our formalism with goals, which seems a natural extension in order to allow the agent to, for instance, replace intentions instead of merely discarding them. This paves the road to develop a richer notion of intentions, such as that "intentions normally pose problems for the agents; the agent needs to determine a way of achieving them" [Cohen and Levesque, 1990]. Interestingly, adding goals to the formalism blurs the distinction between planner and databases. If the databases take over part of the planning, then well-known problems such as the frame problem become more stringent: Once a fact is established (for example, as a postcondition of an intention), it persists until it explicitly contradicts postconditions established by future intentions. Existing action logics (e.g., the Event Calculus or the Fluent Calculus) and database approaches (e.g., TMMS [Dean and McDermott, 1987]) have dealt with these problems in detail, so comparing and possibly enriching them with our formalism seems both useful and relevant future work.

[5] http://www.dropbox.com/s/798g6ki0aqt3gds/ijcai2016proofs.pdf?dl=1

References

[Alchourron *et al.*, 1985] Carlos E. Alchourron, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50(2):510–530, 06 1985.

[Baral and Zhang, 2005] Chitta Baral and Yan Zhang. Knowledge updates: Semantics and complexity issues. *Artificial Intelligence*, 164(1):209–243, 2005.

[Bonanno, 2007] Giacomo Bonanno. Axiomatic characterization of the AGM theory of belief revision in a temporal logic. *Artificial Intelligence*, 171(2):144–160, 2007.

[Bratman, 1987] Michael E. Bratman. *Intention, plans, and practical reason*. Harvard University Press, Cambridge, MA, 1987.

[Cohen and Levesque, 1990] Philip R Cohen and Hector J Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(2-3):213–261, 1990.

[Cohen and Levesque, 1991] Philip R. Cohen and Hector J. Levesque. Teamwork. *Noûs*, 25(4):487–512, 1991.

[Darwiche and Pearl, 1997] Adnan Darwiche and Judea Pearl. On the logic of iterated belief revision. *Artificial Intelligence*, 89(12):1 – 29, 1997.

[Dean and McDermott, 1987] Thomas L Dean and Drew V McDermott. Temporal data base management. *Artificial Intelligence*, 32(1):1–55, 1987.

[Ditmarsch *et al.*, 2011] Hans Ditmarsch, Tiago Lima, and Emiliano Lorini. *Intention Change via Local Assignments*, pages 136–151. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[Fagin *et al.*, 1995] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[Grant *et al.*, 2010] John Grant, Sarit Kraus, Donald Perlis, and Michael Wooldridge. Postulates for revising BDI structures. *Synthese*, 175(1):39–62, 2010.

[Icard *et al.*, 2010] Thomas Icard, Eric Pacuit, and Yoav Shoham. Joint revision of belief and intention. *Proc. of the 12th International Conference on Knowledge Representation*, pages 572–574, 2010.

[Jin and Thielscher, 2004] Yi Jin and Michael Thielscher. Representing beliefs in the fluent calculus. In *ECAI*, pages 823–827. IOS Press, 2004.

[Katsuno and Mendelzon, 1991] Hirofumi Katsuno and Alberto O. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3):263–294, dec 1991.

[Kvarnström, 2005] Jonas Kvarnström. *TALplanner and other extensions to Temporal Action Logic*. PhD thesis, Linköpings universitet, 2005.

[Lorini and Herzig, 2008] Emiliano Lorini and Andreas Herzig. A logic of intention and attempt. *Synthese*, 163(1):45–77, 2008.

[Lorini *et al.*, 2009] Emiliano Lorini, Mehdi Dastani, Hans P. van Ditmarsch, Andreas Herzig, and John-Jules Ch. Meyer. Intentions and assignments. In *LORI*, volume 5834 of *Lecture Notes in Computer Science*, pages 198–211. Springer, 2009.

[McDermott, 1982] Drew McDermott. A temporal logic for reasoning about processes and plans. *Cognitive science*, 6(2):101–155, 1982.

[Mueller, 2010] Erik T Mueller. *Commonsense reasoning*. Morgan Kaufmann, 2010.

[Patkos, 2010] Theodore Patkos. *A formal theory for reasoning about action, knowledge and time*. PhD thesis, University of Crete-Heraklion, 2010.

[Reynolds, 2002] M. Reynolds. An axiomatization of full computation tree logic. *Journal of Symbolic Logic*, 66(3):1011–1057, 2002.

[Scherl and Levesque, 2003] Richard B Scherl and Hector J Levesque. Knowledge, action, and the frame problem. *Artificial Intelligence*, 144(1):1–39, 2003.

[Scherl, 2005] Richard B Scherl. Action, belief change and the frame problem: A fluent calculus approach. In *Proceedings of the Sixth workshop on Nonmonotonic Reasoning, Action, and Change at IJCAI*, 2005.

[Shapiro *et al.*, 2011] Steven Shapiro, Maurice Pagnucco, Yves Lesprance, and Hector J. Levesque. Iterated belief change in the situation calculus. *Artificial Intelligence*, 175(1):165–192, 2011.

[Shoham, 2009] Yoav Shoham. Logical theories of intention and the database perspective. *Journal of Philosophical Logic*, 38(6):633–647, 2009.

[Shoham, 2016] Yoav Shoham. Why knowledge representation matters. *Commun. ACM*, 59(1):47–49, January 2016.

[Thielscher, 2001] Michael Thielscher. The concurrent, continuous fluent calculus. *Studia Logica*, 67(3):315–331, 2001.

[Van der Hoek and Wooldridge, 2003] Wiebe Van der Hoek and Michael Wooldridge. Towards a logic of rational agency. *Logic Journal of IGPL*, 11(2):135–159, 2003.

[van der Hoek *et al.*, 2007] Wiebe van der Hoek, Wojciech Jamroga, and Michael Wooldridge. Towards a theory of intention revision. *Synthese*, 155(2):265–290, 2007.

[van Zee *et al.*, 2015a] Marc van Zee, Mehdi Dastani, Dragan Doder, and Leendert van der Torre. Consistency Conditions for Beliefs and Intentions. In *Twelfth International Symposium on Logical Formalizations of Commonsense Reasoning*, 2015.

[van Zee *et al.*, 2015b] Marc van Zee, Dragan Doder, Mehdi Dastani, and Leendert van der Torre. AGM Revision of Beliefs about Action and Time. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2015.

A New Approach for Revising Logic Programs

Zhiqiang Zhuang[1] **James Delgrande**[2] **Abhaya Nayak**[3] **Abdul Sattar**[1]
[1] Institute for Integrated and Intelligent Systems, Griffith University, Australia
[2] School of Computing Science, Simon Fraser University, Canada
[3] Department of Computing, Macquarie University, Australia

Abstract

Belief revision has been studied mainly with respect to background logics that are monotonic in character. In this paper we study belief revision when the underlying logic is non-monotonic instead—an inherently interesting problem that is under explored. In particular, we will focus on the revision of a body of beliefs that is represented as a logic program under the answer set semantics, while the new information is also similarly represented as a logic program. Our approach is driven by the observation that unlike in a monotonic setting where, when necessary, consistency in a revised body of beliefs is maintained by jettisoning some old beliefs, in a non-monotonic setting consistency can be restored by adding new beliefs as well. We will define a syntactic revision function and subsequently provide representation theorem for characterising it.

Introduction

The ability to change one's beliefs when presented with new information is crucial for any intelligent agent. In the area of *belief change*, substantial effort has been made towards the understanding and realisation of this process. Traditionally, it is assumed that the agent's reasoning is governed by a monotonic logic. For this reason, traditional belief change is inapplicable when the agent's reasoning is non-monotonic. Our goal in this research program is to extend belief base (Hansson 1999) approaches in belief revision to nonmonotonic setting. In this paper, we focus on *disjunctive logic programs*, as a well-studied and well-known approach to non-monotonic reasoning that also has efficient implementations.

Much, if not most, of our day-to-day reasoning involves non-monotonic reasoning. To illustrate issues that may arise, consider the following example. In a university, professors generally teach, unless they have an administrative appointment. Assume we know that John is a professor. Since most faculty do not have an administrative appointment, and there is no evidence that John does, we conclude that he teaches. This reasoning is a classical form of non-monotonic reasoning, namely using the *closed world assumption*. It can be represented by the following logic program under the *answer set semantics*.

$$Teach(X) \leftarrow Prof(X), not\ Admin(X). \tag{1}$$

$$Prof(John) \leftarrow . \tag{2}$$

The *answer set* $\{Prof(John), Teach(John)\}$ for this logic program corresponds exactly to the facts we can conclude.

Suppose we receive information that John does not teach, which we can represent by the rule

$$\leftarrow Teach(John). \tag{3}$$

Now our beliefs about John are contradictory; and it is not surprising that the logic program consisting of rules (1) – (3) has no answer set. For us or any intelligent agent in this situation to function properly, we need a mechanism to resolve this inconsistency. This is a typical belief revision problem; however, the classical (AGM) approach can not be applied, as we are reasoning non-monotonically.

It is not hard to suggest possible causes of the inconsistency and to resolve it. It could be that some of our beliefs are wrong; perhaps professors with administrative duties may still need to do teaching or perhaps John is not a professor. Thus we can restore consistency by removing rule (1) or (2). Alternatively and perhaps more interestingly, it could be that assuming that John is not an administrative staff via the absence of evidence is too adventurous; that is he may indeed be an administrative staff member but we don't know it. Thus we can also restore consistency by adding the missing evidence of John being an administrative staff member by

$$Admin(John) \leftarrow . \tag{4}$$

The second alternative highlights the distinction for belief revision in monotonic and non-monotonic settings. In the monotonic setting, an inconsistent body of knowledge will remain inconsistent no matter how much extra information is supplied. On the other hand, in the non-monotonic setting, inconsistency can be resolved by either removing old information, or adding new information, or both. Therefore, belief revision functions in a non-monotonic setting should allow a mixture of removal and addition of information for inconsistency-resolution. In this paper, we will define one such revision functions for disjunctive logic programs under the answer set semantics.

The revision function is called *slp-revision* and is a belief base revision which takes syntactic information into account. In revising P by Q, an slp-revision function first obtains a logic program R that is consistent with Q and differs minimally from P, then combines R with Q. For example, if $P = \{(1), (2)\}$ and $Q = \{(3)\}$, then R could be $\{(1)\}$ (i.e., resolving inconsistency by removing (2)); $\{(2)\}$ (i.e., resolving inconsistency by removing (1)); or $\{(1), (2), (4)\}$ (i.e., resolving inconsistency by adding (4)).

The next section gives logical preliminaries. The following one develop our approach to slp-revision in which we provide postulates, a semantic construction, and a representation result. This is followed by a comparison to other work, and a brief conclusion.

Preliminary Considerations

In this paper, we consider only fully grounded disjunctive logic programs. That is variables in program rules are replaced by the set of their ground instances. Thus a logic program (or program for short) here is a finite set of rules of the form:

$$a_1; \dots; a_m \leftarrow b_1, \dots, b_n, not\ c_1, \dots, not\ c_o$$

where $m, n, o \geq 0$, $m + n + o > 0$, and $a_i, b_j, c_k \in \mathcal{A}$ for $\mathcal{A}$ a finite set of propositional atoms. Connective *not* is called *default negation*. We denote the set of all logic programs by $\mathcal{P}$. For each rule r, let $H(r) = \{a_1, \dots, a_n\}$, $B^+(r) = \{b_1, \dots, b_m\}$, and $B^-(r) = \{c_1, \dots, c_o\}$. The letters P and Q are used to denote a logic program throughout the paper.

An interpretation is represented by the subset of atoms in $\mathcal{A}$ that are true in the interpretation. A *classical model* of a program P is an interpretation in which all rules of P are true according to the standard definition of truth in propositional logic, and where default negation is treated as classical negation. The set of classical models of P is denoted as $Mod(P)$. Given an interpretation Y, we write $Y \models P$ to mean Y is a classical model of P. The *reduct* of a program P with respect to an interpretation Y, denoted P^Y, is the set of rules:

$$\{H(r) \leftarrow B^+(r) \mid r \in P, B^-(r) \cap Y = \emptyset\}.$$

An *answer set* Y of P is a subset-minimal classical model of P^Y. The set of all answer set of P is denoted as $AS(P)$.

An *SE interpretation* (Turner 2003) is a pair (X, Y) of interpretations such that $X \subseteq Y \subseteq \mathcal{A}$. The set of all SE interpretations (over $\mathcal{A}$) is denoted $\mathcal{SE}$. The letters M and N are used to denote a set of SE interpretations throughout the paper. An SE interpretation is an *SE model* of a program P if $Y \models P$ and $X \models P^Y$. The set of all SE models of P is denoted as $SE(P)$. SE models are proposed to capture *strong equivalence* (Lifschitz *et al.* 2001) between programs that is $SE(P) = SE(Q)$ iff P and Q are strongly equivalent, thus they contain more informations than answer sets.

The following two properties of SE models (Turner 2003) are crucial to this paper:

1. $Y \in AS(P)$ iff $(Y, Y) \in SE(P)$ and there is no $(X, Y) \in SE(P)$ such that $X \subset Y$.
2. $(Y, Y) \in SE(P)$ iff $Y \in Mod(P)$.

So $SE(P) \neq \emptyset$ iff $Mod(P) \neq \emptyset$ but $SE(P) \neq \emptyset$ does not imply $AS(P) \neq \emptyset$. This gives rise to two notions of consistency.

Definition 1. *P is* consistent *iff $AS(P) \neq \emptyset$ and P is* m-consistent[1] *iff $SE(P) \neq \emptyset$.*

It is clear from the SE model properties that consistency implies m-consistency; m-inconsistency implies inconsistency. In other words, a consistent program is m-consistent but not vice versa.

In subsequent sections, we will need to describe the difference between two logic programs. For this purpose, we use the *symmetric difference* operator $\ominus$ which is defined as

$$X \ominus Y = (X \setminus Y) \cup (Y \setminus X)$$

for any sets X and Y.

SLP-Revision Functions

In this section, we give a syntax-based revision function $* : \mathcal{P} \times \mathcal{P} \mapsto \mathcal{P}$ for revising one logic program by another. The function takes a logic program P called the *original logic program* and a logic program Q called the *revising logic program*, and returns another logic program $P * Q$ called the *revised logic program*. Following AGM belief revision, we want to have Q contained in $P * Q$ (i.e., $Q \subseteq P * Q$), $P * Q$ is consistent whenever possible, and that as much of P as consistently possible is contained in $P * Q$.

Clearly, a key issue in defining $*$ is to deal with the possible inconsistency between Q and P. As illustrated in the teaching example, one means of ensuring that $P * Q$ is consistent is to remove a minimal set of beliefs from P so that adding Q to the result is consistent. Of course there may be more than one way to remove beliefs from P. Following this intuition, we obtain all maximal subsets of P that are consistent with Q, which we call the *s-removal compatible programs* of P with respect to Q.

Definition 2. *The set of* s-removal compatible programs *of P with respect to Q, denoted $P \downarrow Q$, is such that $R \in P \downarrow Q$ iff*
1. $R \subseteq P$,
2. $R \cup Q$ is consistent, and
3. if $R \subset R' \subseteq P$, then $R' \cup Q$ is inconsistent.

The notion of s-removal compatible programs is not new, classical revision functions (Alchourrón *et al.* 1985; Hansson 1993) are based on more or less the same notion. The difference is that this notion alone is sufficient to capture the inconsistency-resolution strategy of classical belief revision, but there is more that one can do in non-monotonic belief revision.

In our non-monotonic setting, we are able to express assumptions (i.e., negation as failure) and to reason with them. Earlier, we assumed John is not an administrator, in the absence of evidence to the contrary. With this, we came to the conclusion that he has to teach. Consequently, if we learn

[1] "m" stands for "monotonic" which indicates that the notion of m-consistency is based on a monotonic characterisation (i.e., SE models) for logic programs.

that John does not teach, as in our example, one way of resolving this inconsistency is by adding information so that our assumption does not hold. Following this intuition, we obtain all the minimal supersets of P that are consistent with Q, which we call the *s-expansion compatible program* of P with respect to Q.

Definition 3. *The set of* s-expansion compatible programs *of P with respect to Q, denoted $P \uparrow Q$, is such that $R \in P \uparrow Q$ iff*
1. $P \subseteq R$,
2. $R \cup Q$ is consistent, and
3. if $P \subseteq R' \subset R$, then $R' \cup Q$ is inconsistent.

Since the s-expansion and s-removal compatible programs are consistent with Q and are obtained by removing or adding minimal sets of rules from or to P, the union of Q with any of these sets is consistent and comprises a least change made to P in order to achieve consistency. These programs clearly should be candidates for forming the revised logic program $P * Q$; however, they do not form the set of all candidates. In particular, we can obtain a program that differs the least from P and is consistent with Q by removing some beliefs of P and at the same time adding some new beliefs to P. Thus we consider all those logic programs that differ the least from P and are consistent with Q; these are called the *s-compatible programs* of P with respect to Q.

Definition 4. *The set of* s-compatible programs *of P with respect to Q, denoted $P \updownarrow Q$, is such that $R \in P \updownarrow Q$ iff*
1. $R \cup Q$ is consistent and
2. if $P \ominus R' \subset P \ominus R$, then $R' \cup Q$ is inconsistent.

For example, let $P = \{a \leftarrow b, not\, c., b., e \leftarrow f, not\, g., f.\}$ and $Q = \{\leftarrow a., \leftarrow e.\}$. Then $P \cup Q$ is inconsistent since a and e can be concluded from P but they contradict the rules of Q. To resolve the inconsistency via making the least change to P, we could remove $b \leftarrow$ from P (which eliminates the contradiction about a) and add $g \leftarrow$ to P (which eliminates the contradiction about e). The program thus obtained (i.e., $(P \setminus \{b.\}) \cup \{g.\}$) is a s-compatible program in $P \updownarrow Q$.

It is obvious, but worth noting that the notion of s-compatible program subsumes those of s-removal and s-expansion compatible programs. In the above example, $P \updownarrow Q$ also contains $P \setminus \{b., f.\}$ and $P \cup \{c., g.\}$, which are respectively an s-removal and an s-expansion compatible program of P with respect to Q.

Proposition 1. $(P \uparrow Q) \cup (P \downarrow Q) \subseteq P \updownarrow Q$.

There are cases in which we cannot resolve inconsistency by only adding new beliefs which means the set of s-expansion compatible programs is empty. For example, if $P = \{a.\}$ and $Q = \{\leftarrow a.\}$, then $P \cup Q$ is inconsistent and we cannot restore consistency without removing $a \leftarrow$ from P. In these cases, the inconsistency is due to contradictory facts that can be concluded without using any reasoning power beyond that of classical logic. Clearly, the inconsistency is of a monotonic nature, that is, in our terminology, m-inconsistency.

Proposition 2. *If $P \cup Q$ is m-inconsistent, then $P \uparrow Q = \emptyset$.*

So far, we have identified the candidates for forming $P * Q$. It remains to pick the "best" one. Such extralogical information is typically modelled by a *selection function*, which we do next.

Definition 5. *A function γ is a* selection function *for P iff for any program Q, $\gamma(P \updownarrow Q)$ returns a single element of $P \updownarrow Q$ whenever $P \updownarrow Q$ is non-empty; otherwise it returns P.*

The revised logic program $P * Q$ is then formed by combining Q with the s-compatible program picked by the selection function for P. We call the function $*$ defined in this way a *slp-revision function* for P.

Definition 6. *A function $*$ is a slp-revision function for P iff*

$$P * Q = \gamma(P \updownarrow Q) \cup Q$$

for any program Q, where γ is a selection function for P.

In classical belief revision, multiple candidates maybe chosen by a selection function, and their intersection is combined with the new belief to form the revision result. There, a selection function that picks out a single element is called a *maxichoice* function (Alchourrón *et al.* 1985). In classical logic, maxichoice selection functions leads to undesirable properties for belief set revision but not for belief base revision. In our non-monotonic setting, picking multiple candidates does not make sense, as intersection of s-compatible programs may not be consistent with the revising program. For example, let $P = \{a \leftarrow not\, b, not\, c.\}$ and $Q = \{\leftarrow a.\}$. We can restore consistency of P with Q by, for instance, adding the rule $b \leftarrow$ to P which corresponds to the s-compatible program $P \cup \{b.\}$ or by adding the rule $c \leftarrow$ which corresponds to the s-compatible program $P \cup \{c.\}$. However, the intersection of the two s-compatible programs is inconsistent with Q.

We turn next to properties of slp-revision functions. Consider the following set of postulates where $* : \mathcal{P} \times \mathcal{P} \mapsto \mathcal{P}$ is a function.

(s$*$s) $Q \subseteq P * Q$
(s$*$c) If Q is m-consistent, then $P * Q$ is consistent
(s$*$f) If Q is m-inconsistent, then $P * Q = P \cup Q$
(s$*$rr) If $R \neq \emptyset$ and $R \subseteq P \setminus (P * Q)$, then
$(P * Q) \cup R$ is inconsistent
(s$*$er) If $E \neq \emptyset$ and $E \subseteq (P * Q) \setminus (P \cup Q)$, then
$(P * Q) \setminus E$ is inconsistent
(s$*$mr) If $R \neq \emptyset$, $R \subseteq P \setminus (P * Q)$,
$E \neq \emptyset$ and $E \subseteq (P * Q) \setminus (P \cup Q)$, then
$((P * Q) \cup R) \setminus E$ is inconsistent
(s$*$u) If $P \updownarrow Q = P \updownarrow R$, then
$P \setminus (P * Q) = P \setminus (P * R)$ and
$(P * Q) \setminus (P \cup Q) = (P * R) \setminus (P \cup R)$

(s$*$s) (*Success*) states that a revision is always successful in incorporating the new beliefs. (s$*$c) (*Consistency*) states that a revision ensures consistency of the revised logic program whenever possible. In the monotonic setting, a revision results in inconsistency only when the new beliefs are themselves inconsistent. This is not the case in the non-monotonic setting. For example, consider the revision of $P = \{a.\}$ by $Q = \{b \leftarrow not\ b\}$. Although Q is inconsistent, we have $P \cup \{b.\}$ as a s-compatible program of P

with respect to Q. Thus we can have $P \cup \{b.\} \cup Q$ as the revised logic program, which contains Q and is consistent. Here, a revision results in inconsistency only when the revising logic program is m-inconsistent. In such a case, (s$*$f) (*Failure*) states that the revision corresponds to the union of the original and revising logic program.

(s$*$rr) (*Removal Relevance*) states that if some rules are removed from the original logic program for the revision, then adding them to the revised logic program results in inconsistency. It captures the intuition that nothing is removed unless its removal contributes to making the revised logic program consistent. (s$*$er) (*Expansion Relevance*) states that if some new rules other than those in the revising logic program are added to the original logic program for the revision, then removing them from the revised logic program causes inconsistency. It captures the intuition that nothing is added unless adding it contributes to making the revised logic program consistent. (s$*$mr) (*Mixed Relevance*) states that if some rules are removed from the original logic program and some new rules other than those in the revising logic program are added to the original logic program for the revision, then adding back the removed ones and removing the added ones result in inconsistency of the revised logic program. Its intuition is a mixture of the two above. Note that putting (s$*$rr) and (s$*$er) together does not guarantee (s$*$mr), nor the reverse. In summary, these three postulates express the necessity of adding and/or removing certain belief for resolving inconsistency and hence to accomplish a revision. In classical belief revision, inconsistency can only be resolved by removing old beliefs; the necessity of removing particular beliefs is captured by the *Relevance* postulate (Hansson 1993).[2] The three postulates are the counterparts of *Relevance* in our non-monotonic setting, and we need all three of them to deal respectively with addition, removal, and a mixture of addition and removal.

Finally, (s$*$u) (*Uniformity*) states the condition under which two revising logic programs Q and R trigger the same changes to the original logic program P. That is the rules removed from P (i.e., $P \setminus (P * Q)$) and the rules added to P (i.e., $(P * Q) \setminus (P \cup Q)$) for accommodating Q are identical to those for accommodating R. Certainly having Q and R be strongly equivalent (i.e., $SE(Q) = SE(R)$) is a sufficient condition. However, it is too strong a requirement. Suppose $P = \{\leftarrow a.\}$, $Q = \{a.\}$, and $R = \{a \leftarrow b., b.\}$. Then the minimal change to P we have to made to accommodate Q and R are the same, that is we remove $\leftarrow a$. However Q and R are not strongly equivalent, even though they incur the same change to P. The essential point of this example is that instead of a global condition like strong equivalence, we need a condition that is local to the original logic program P. Unfortunately, it seems there is no existing notion in the logic programming literature that captures this local condition. Thus we use our newly defined notion of s-compatible programs and come up with the local but more appropriate condition in (s$*$u).

We can show that these postulates are sufficient to characterise all slp-revision functions.

Theorem 1. *A function $*$ is a slp-revision function iff it satisfies (s$*$s), (s$*$c), (s$*$f), (s$*$rr), (s$*$er), (s$*$mr), and (s$*$u).*

Comparisons with Existing Approaches

There has been much work on belief revision for logic programs. The seminal work of Delgrande et al (2013b) generalises Satoh's (1988) and Dalal's (1988) revision operators to logic programs. Significantly, they bring SE model into the picture. They do not work with answer sets as a basis for revision, but rather they base their definitions directly on SE models. The work has inspired several other SE model approaches. Schwind and Inoue (2013) provide a constructive characterisation for the revision operators in (Delgrande *et al.* 2013b). Delgrande et al (2013a) adapt the model-based revision of Katsuno and Mendelzon (1992) to logic programs and provide a representation theorem. Finally, Binnewies et al (2015) provide a variant of partial meet revision and contraction for logic programs.

Firstly, the SE model approaches are essentially belief set revision whereas our slp-revision is a belief base one. Secondly and more importantly, these approaches assume a weaker notion of consistency, that is m-consistency. For this reason, some contradictions will not be dealt with in these approaches. For instance, the contradictory rule $a \leftarrow not\, a$ is m-consistent thus is considered to be an acceptable state of belief. Also in our teaching example, as the program consisting of rules (1) – (3) is m-consistent, no attempt will be made to resolve the contradiction about John's teaching duty by the SE model approaches. Therefore for application scenarios in which such contradictions can not be tolerant, our llp-revision function is clearly a better choice.

Apart from the SE model approaches, Krümpelmann and Kern-Isberner (2012) provide a revision function for logic programs that originates from Hansson's *semi-revision* (Hansson 1997). Since they assume the same notion of consistency as ours, all the above mentioned contradictions will be resolved in their approach.

As we have noted, classical belief revision is defined for monotonic setting, not for non-monotonic ones. Inconsistency can be caused by wrong assumptions in the non-monotonic setting but not in the monotonic setting. Such causes are not considered in (Krümpelmann and Kern-Isberner 2012). Consequently, their approach only support one of the many possible inconsistency-resolution strategies we have developed. Specifically, in (Krümpelmann and Kern-Isberner 2012), inconsistency can be resolved only by removing old beliefs; this strategy is captured by a notion analogous to s-removal compatible programs. The inconsistency-resolution strategies captured by the notion of s-expansion compatible program and s-compatible program in general are not considered.

Conclusion and Future Work

Depending on the application scenario, the logic governing an agent's beliefs could be either monotonic or non-monotonic. Traditional belief revision assumes that an agent

[2] If $\psi \in K$ and $\psi \notin K * \phi$, then there is some K' such that $K * \phi \subseteq K' \subseteq K \cup \{\phi\}$, K' is consistent but $K' \cup \{\psi\}$ is inconsistent.

reasons monotonically; therefore, by definition, it is applicable to such situations only. Here we have aimed to provide a belief revision framework for situations in which the agent reasons non-monotonically. To this end, we defined a belief revision function for disjunctive logic programs under the answer set semantics.

Inconsistency-resolution is an essential task for belief revision. However, the strategies used in traditional belief revision functions are limited to situations when the agent reasons monotonically. With a logic program we have the luxury of making assumptions via lack of contrary evidence, and we can deduce certain facts from such assumptions. Thus if a set of beliefs is inconsistent, then one possible cause is that we made the wrong assumption. In such cases, we can resolve the inconsistency by adding some new rules so that the assumption can no longer be made. Such a cause of inconsistency and the associated inconsistency-resolution strategy is beyond the scope of traditional belief revision, but is crucial for non-monotonic belief revision. We argue that this rationale, which is encoded in our belief revision function, captures the fundamental difference between monotonic and non-monotonic belief revision.

This paper then has explored belief base revision in the non-monotonic setting of disjunctive logic programs. Note that the characterising postulates of the base revision are formulated in terms of set-theoretic notions (e.g., subsets, set differences); the only logical notion required is consistency. Moreover the key idea, namely the notion of s-compatible programs, is also based on the same set-theoretic and logical notions. These notions are present in all non-monotonic settings. In future work we propose to extend the base revision to a general approach to belief revision in arbitrary non-monotonic settings.

Appendix: Proof of Results

In this appendix, we give the proof for the main results.

Proof for Proposition 2

Let P and Q are logic programs. Suppose $P \cup Q$ is m-inconsistent. We need to show $P \uparrow Q = \emptyset$.

Since $P \cup Q$ is m-inconsistent, we have $SE(P) \cap SE(Q) = \emptyset$. By the definition of s-expansion compatible program, any element in $P \uparrow Q$ has to be a superset of P and consistent with Q. However, for any superset R of P, we have $SE(R) \subseteq SE(P)$. Thus $SE(R) \cap SE(Q) = \emptyset$ which implies $R \cup Q$ is m-inconsistent.

□

Proof for Theorem 1

For one direction, suppose $*$ is a slp-revision function for P and the associated selection function is γ. We need to show $*$ satisfies (s$*$s), (s$*$c), (s$*$f), (s$*$rr), (s$*$er), (s$*$mr), and (s$*$u). (s$*$s), (s$*$c), and (s$*$f) follow immediately from the definition of slp-revision functions and compatible programs.

(s$*$rr): Suppose there is a set R such that $R \neq \emptyset$ and $R \subseteq P \setminus (P * Q)$. By the definition of slp-revision, we have $P * Q = \gamma(P \updownarrow Q) \cup Q$, hence $P \setminus (\gamma(P \updownarrow Q) \cup Q) \neq \emptyset$ which implies $\gamma(P \updownarrow Q) \neq P$. Then it follows from the definition of selection function that $P \updownarrow Q \neq \emptyset$ and $\gamma(P \updownarrow Q) \in P \updownarrow Q$. Let $\gamma(P \updownarrow Q) = X$. Then $(P*Q) \cup R = X \cup Q \cup R$. Since $\emptyset \neq R \subseteq P$, we have $((X \cup R) \ominus P) \subset (X \ominus P)$. By the definition of compatible program, $X \cup R \cup Q$ is inconsistent that is $(P * Q) \cup R$ is inconsistent.

(s$*$er): Suppose there is a set E such that $E \neq \emptyset$ and $E \subseteq (P * Q) \setminus (P \cup Q)$. By the definition of slp-revision, we have $P * Q = \gamma(P \updownarrow Q) \cup Q$, hence $(\gamma(P \updownarrow Q) \cup Q) \setminus (P \cup Q) \neq \emptyset$ which implies $\gamma(P \updownarrow Q) \neq P$. Then it follows from the definition of selection function that $P \updownarrow Q \neq \emptyset$ and $\gamma(P \updownarrow Q) \in P \updownarrow Q$. Let $\gamma(P \updownarrow Q) = X$. Then $(P*Q) \setminus E = (X \cup Q) \setminus E$. Since $E \cap P = \emptyset$ and $\emptyset \neq E \subseteq X$, $((X \setminus E) \ominus P) \subset (X \ominus P)$. By the definition of compatible program, $(X \setminus E) \cup Q$ is inconsistent. Then since $E \cap Q = \emptyset$, we have $(X \setminus E) \cup Q = (X \cup Q) \setminus E = (P * Q) \setminus E$. Thus $(P * Q) \setminus E$ is inconsistent.

(s$*$mr): Can be proved by combining the proving method for (s$*$rr) and (s$*$er).

(s$*$u): Suppose $P \updownarrow Q = P \updownarrow R$. Then $\gamma(P \updownarrow Q) = \gamma(P \updownarrow R)$. If $P \updownarrow Q = P \updownarrow R = \emptyset$, then by the definition of slp-revision $P * Q = P \cup Q$ and $P * R = P \cup R$. Thus $P \setminus (P * Q) = P \setminus (P * R) = \emptyset$ and $(P * Q) \setminus (P \cup Q) = (P * R) \setminus (P \cup R) = \emptyset$. So suppose $P \updownarrow Q = P \updownarrow R \neq \emptyset$ and let $X = \gamma(P \updownarrow Q) = \gamma(P \updownarrow R)$. By the definition of slp-revision, we have $P \setminus (P * Q) = P \setminus (X \cup Q)$. Assume $\emptyset \neq P \cap Q \not\subseteq X$. Then since $X \cup (P \cap Q)$ is consistent with Q and $(X \cup (P \cap Q)) \ominus P \subset X \ominus P$, X is not a compatible program, a contradiction! Thus $P \cap Q = \emptyset$ or $P \cap Q \subseteq X$. In either case we have by set theory that $P \setminus (P * Q) = P \setminus (X \cup Q) = P \setminus X$. It can be shown in the same manner that $P \setminus (P * R) = P \setminus (X \cup R) = P \setminus X$. Thus $P \setminus (P * Q) = P \setminus (P * R)$. Again by the definition of slp-revision, we have $(P * Q) \setminus (P \cup Q) = (X \cup Q) \setminus (P \cup Q) = X \setminus P$. Similarly $(P * R) \setminus (P \cup R) = (X \cup R) \setminus (P \cup R) = X \setminus P$. Thus $(P * Q) \setminus (P \cup Q) = (P * R) \setminus (P \cup R)$.

For the other direction, suppose $*$ is a function that satisfies (s$*$s), (s$*$c), (s$*$f), (s$*$rr), (s$*$er), (s$*$mr), and (s$*$u). We need to show $*$ is a slp-revision function.

Let γ be defined as:

$$\gamma(P \updownarrow Q) = ((P * Q) \cap P) \cup ((P * Q) \setminus Q)$$

for all Q. It suffices to show γ is a selection function for P and $P * Q = \gamma(P \updownarrow Q) \cup Q$.

Part 1: For γ to be a selection function, it must be a function. Suppose $P \updownarrow Q = P \updownarrow R$. Then (s$*$u) implies $P \setminus (P * Q) = P \setminus (P * R)$ and $(P * Q) \setminus (P \cup Q) = (P * R) \setminus (P \cup R)$. Since $P = (P \setminus (P * Q)) \cup ((P * Q) \cap P) = (P \setminus (P * R)) \cup ((P * R) \cap P)$, $P \setminus (P * Q) = P \setminus (P * R)$ implies $(P * Q) \cap P = (P * R) \cap P$. Thus $(P * Q) \setminus (P \cup Q) = (P * R) \setminus (P \cup R)$ implies $((P * Q) \cap P) \cup ((P * Q) \setminus (P \cup Q)) = ((P * R) \cap P) \cup ((P * R) \setminus (P \cup R))$. Then by set theory, we have $((P * Q) \cap P) \cup ((P * Q) \setminus Q) = ((P * R) \cap P) \cup ((P * R) \setminus R)$. Finally, it follows from the definition of γ that $\gamma(P \updownarrow Q) = \gamma(P \updownarrow R)$.

If $P \updownarrow Q = \emptyset$, then we have to show $\gamma(P \updownarrow Q) = P$. $P \updownarrow Q = \emptyset$ implies Q is m-inconsistent, hence it follows from (s$*$f) that $P * Q = P \cup Q$. Then by the definition of γ, $\gamma(P \updownarrow Q) = ((P * Q) \cap P) \cup ((P * Q) \setminus Q) = ((P \cup Q) \cap P) \cup ((P \cup Q) \setminus Q) = P$.

If $P \updownarrow Q \neq \emptyset$, then we have to show $\gamma(P \updownarrow Q) \in P \updownarrow Q$. Since $P \updownarrow Q \neq \emptyset$, Q is m-consistent. Then (s$*$c) implies $P * Q$ is consistent. Since $\gamma(P \updownarrow Q) \cup Q = ((P*Q) \cap P) \cup ((P*Q) \setminus Q) \cup Q = P*Q$, $\gamma(P \updownarrow Q) \cup Q$ is consistent. Assume there is X s.t. $X \cup Q$ is consistent and $X \ominus P \subset \gamma(P \updownarrow Q) \ominus P$. Then we have three cases:

Case 1, there is R s.t. $\emptyset \neq R \subseteq P \setminus \gamma(P \updownarrow Q)$, and $X = \gamma(P \updownarrow Q) \cup R$: If $R \cap Q = \emptyset$, then since $\gamma(P \updownarrow Q) \cup Q = P*Q$, $R \cap (P*Q) = \emptyset$. Then it follows from (s$*$rr) that $(P*Q) \cup R$ is inconsistent. Since $X \cup Q = (P*Q) \cup R$, $X \cup Q$ is inconsistent, a contradiction! If $R \cap Q \neq \emptyset$, then since $R \subseteq P$, $R \cap P \cap Q \neq \emptyset$. Since (s$*$s) implies $Q \subseteq P*Q$, we have $Q \cap P \subseteq (P*Q) \cap P$, which implies $R \cap ((P*Q) \cap P) \neq \emptyset$. Then since $((P*Q) \cap P) \subseteq \gamma(P \updownarrow Q)$, $\gamma(P \updownarrow Q) \cap R \neq \emptyset$, a contradiction! Thus $R \cap Q \neq \emptyset$ is an impossible case.

Case 2, there is E s.t. $E \cap P = \emptyset$, $\emptyset \neq E \subseteq \gamma(P \updownarrow Q)$, and $X = \gamma(P \updownarrow Q) \setminus E$: Then $E \subseteq \gamma(P \updownarrow Q) \cup Q = P*Q$. If $E \cap Q = \emptyset$, then (s$*$er) implies $(P*Q) \setminus E$ is inconsistent. Since $X \cup Q = \gamma(P \updownarrow Q) \setminus E \cup Q = (P*Q) \setminus E$, $X \cup Q$ is inconsistent, a contradiction! If $E \cap Q \neq \emptyset$, then $E \not\subseteq (P*Q) \setminus Q$. Since $E \cap P = \emptyset$, we have $E \cap (P*Q) \cap P = \emptyset$. Thus $E \not\subseteq ((P*Q) \cap P) \cup ((P*Q) \setminus Q) = \gamma(P \updownarrow Q)$, a contradiction! Thus $E \cap Q \neq \emptyset$ is an impossible case.

Case 3, there are R and E s.t. $\emptyset \neq R \subseteq P$, $R \cap \gamma(P \updownarrow Q) = \emptyset$, $E \cap P = \emptyset$, $\emptyset \neq E \subseteq \gamma(P \updownarrow Q)$, and $X = (\gamma(P \updownarrow Q) \cup R) \setminus E$: Then we can show as in Case 1 and 2 that $R \cap P*Q = \emptyset$ and $E \subseteq P*Q$. If $R \cap Q = \emptyset$ and $E \cap Q = \emptyset$, then (s$*$mr) implies $((P*Q) \cup R) \setminus E$ is inconsistent. Thus $X \cup Q = ((\gamma(P \updownarrow Q) \cup R) \setminus E) \cup Q = ((P*Q) \cup R) \setminus E$ is inconsistent, a contradiction! Also we can show as in Case 1 and 2 that that $R \cap Q = \emptyset$ and $E \cap Q = \emptyset$ are impossible cases.

Part 2: By set theory, $\gamma(P \updownarrow Q) \cup Q = ((P*Q) \cap P) \cup ((P*Q) \setminus Q) \cup Q = ((P*Q) \cap P) \cup (P*Q) = P*Q$. □

References

Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2):510–530, 1985.

Sebastian Binnewies, Zhiqiang Zhuang, and Kewen Wang. Partial meet revision and contraction in logic programs. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI-2015)*, 2015.

Mukesh Dalal. Investigations into a theory of knowledge base revision. In *Proceedings of the 7th National Conference on Artificial Intelligence (AAAI-1988)*, pages 475–479, 1988.

James P. Delgrande, Pavlos Peppas, and Stefan Woltran. Agm-style belief revision of logic programs under answer set semantics. In *Proceedings of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR-2013)*, pages 264–276, 2013.

James P. Delgrande, Torsten Schaub, Hans Tompits, and Stefan Woltran. A model-theoretic approach to belief change in answer set programming. *ACM Trans. Comput. Log.*, 14(2), 2013.

Sven Ove Hansson. Reversing the Levi Identity. *Journal of Philosophical Logic*, 22(6):637–669, 1993.

Sven Ove Hansson. Semi-revision. *Journal of Applied Non-Classical Logics*, 7(1-2):151–175, 1997.

Sven Ove Hansson. *A Textbook of Belief Dynamics Theory Change and Database Updating*. Kluwer, 1999.

Hirofumi Katsuno and Alberto O. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3):263–294, 1992.

Patrick Krümpelmann and Gabriele Kern-Isberner. Belief base change operations for answer set programming. In *Logics in Artificial Intelligence - 13th European Conference, JELIA 2012, Toulouse, France, September 26-28, 2012. Proceedings*, pages 294–306, 2012.

Vladimir Lifschitz, David Pearce, and Agustín Valverde. Strongly equivalent logic programs. *ACM Trans. Comput. Logic*, 2(4):526–541, 2001.

Ken Satoh. Nonmonotonic reasoning by minimal belief revision. In *Proceedings of the International Conference on Fifth Generation Computer Systems*, pages 455–462, 1988.

Nicolas Schwind and Katsumi Inoue. Characterization theorems for revision of logic programs. In *Proceedings of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR-2013)*, pages 485–498, 2013.

Hudson Turner. Strong equivalence made easy: Nested expressions and weight constraints. *Theory Pract. Log. Program.*, 3(4):609–622, 2003.

Forschungsberichte
der Fakultät für Informatik
der Technischen Universität Dortmund

ISSN 0933-6192

Anforderungen an:
Dekanat Informatik | TU Dortmund
D-44221 Dortmund