

Frontiers  
in  
Artificial  
Intelligence  
and  
Applications

# COMPUTATIONAL MODELS OF ARGUMENT

Proceedings of COMMA 2008

Edited by  
Philippe Besnard  
Sylvie Doutre  
Anthony Hunter

IOS  
Press

# **COMPUTATIONAL MODELS OF ARGUMENT**

# Frontiers in Artificial Intelligence and Applications

FAIA covers all aspects of theoretical and applied artificial intelligence research in the form of monographs, doctoral dissertations, textbooks, handbooks and proceedings volumes. The FAIA series contains several sub-series, including “Information Modelling and Knowledge Bases” and “Knowledge-Based Intelligent Engineering Systems”. It also includes the biennial ECAI, the European Conference on Artificial Intelligence, proceedings volumes, and other ECCAI – the European Coordinating Committee on Artificial Intelligence – sponsored publications. An editorial panel of internationally well-known scholars is appointed to provide a high quality selection.

## Series Editors:

J. Breuker, R. Dieng-Kuntz, N. Guarino, J.N. Kok, J. Liu, R. López de Mántaras,  
R. Mizoguchi, M. Musen, S.K. Pal and N. Zhong

## Volume 172

*Recently published in this series*

- Vol. 171. P. Wang et al. (Eds.), Artificial General Intelligence 2008 – Proceedings of the First AGI Conference
- Vol. 170. J.D. Velásquez and V. Palade, Adaptive Web Sites – A Knowledge Extraction from Web Data Approach
- Vol. 169. C. Branki et al. (Eds.), Techniques and Applications for Mobile Commerce – Proceedings of TAMoCo 2008
- Vol. 168. C. Riggelsen, Approximation Methods for Efficient Learning of Bayesian Networks
- Vol. 167. P. Buitelaar and P. Cimiano (Eds.), Ontology Learning and Population: Bridging the Gap between Text and Knowledge
- Vol. 166. H. Jaakkola, Y. Kiyoki and T. Tokuda (Eds.), Information Modelling and Knowledge Bases XIX
- Vol. 165. A.R. Lodder and L. Mommers (Eds.), Legal Knowledge and Information Systems – JURIX 2007: The Twentieth Annual Conference
- Vol. 164. J.C. Augusto and D. Shapiro (Eds.), Advances in Ambient Intelligence
- Vol. 163. C. Angulo and L. Godo (Eds.), Artificial Intelligence Research and Development
- Vol. 162. T. Hirashima et al. (Eds.), Supporting Learning Flow Through Integrative Technologies
- Vol. 161. H. Fujita and D. Pisanelli (Eds.), New Trends in Software Methodologies, Tools and Techniques – Proceedings of the sixth SoMeT\_07
- Vol. 160. I. Maglogiannis et al. (Eds.), Emerging Artificial Intelligence Applications in Computer Engineering – Real World AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies
- Vol. 159. E. Tyugu, Algorithms and Architectures of Artificial Intelligence

# Computational Models of Argument

Proceedings of COMMA 2008

Edited by

Philippe Besnard

*CNRS, IRIT, Université Toulouse 3, France*

Sylvie Doutre

*IRIT, Université Toulouse 1, France*

and

Anthony Hunter

*Department of Computer Science, University College London, UK*

**IOS**  
Press

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

© 2008 The authors and IOS Press.

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without prior written permission from the publisher.

ISBN 978-1-58603-859-5

Library of Congress Control Number: 2008925574

*Publisher*

IOS Press  
Nieuwe Hemweg 6B  
1013 BG Amsterdam  
Netherlands  
fax: +31 20 687 0019  
e-mail: [order@iospress.nl](mailto:order@iospress.nl)

*Distributor in the UK and Ireland*

Gazelle Books Services Ltd.  
White Cross Mills  
Hightown  
Lancaster LA1 4XS  
United Kingdom  
fax: +44 1524 63232  
e-mail: [sales@gazellebooks.co.uk](mailto:sales@gazellebooks.co.uk)

*Distributor in the USA and Canada*

IOS Press, Inc.  
4502 Rachael Manor Drive  
Fairfax, VA 22032  
USA  
fax: +1 703 323 3668  
e-mail: [iosbooks@iospress.com](mailto:iosbooks@iospress.com)

**LEGAL NOTICE**

The publisher is not responsible for the use which might be made of the following information.

**PRINTED IN THE NETHERLANDS**

## Preface

Models of argumentation, that take pros and cons for some conclusion into account, have been extensively studied over a number of years, and some basic principles have now been clearly established. More recently, there has been interest in computational models of argumentation where the aim is to develop software tools to assist users in constructing and evaluating arguments and counterarguments and/or to develop automated systems for constructing and evaluating arguments and counterarguments.

After the very successful *First Conference on Computational Models of Argument* that was instigated by the members of the EU-funded ASPIC Project, and hosted by the University of Liverpool in September 2006, the papers in this volume form the programme for the *Second Conference on Computational Models of Argument* hosted by the Institut de Recherche en Informatique de Toulouse (IRIT) in May 2008.

The range of papers in the volume provides a valuable snapshot of the leading research questions in the area of computational models of argument. This volume includes papers drawing on the wealth of research on the philosophical questions surrounding the notions arising in argumentation, papers that are addressing knowledge representation and reasoning issues emerging from modelling argumentation, and papers that are considering appropriate models of wider rhetorical issues arising in argumentation. This volume also includes papers that are proposing and evaluating algorithmic solutions associated with generating and judging constellations of arguments, and papers that are proposing standards for exchanging information associated with argumentation so that different systems can work together. We are also pleased to include some papers that report practical working tools for computational argumentation.

We would like to thank the programme committee who did an excellent job in selecting the 37 papers in this volume from the 60 papers that were originally submitted. We would also like to thank the Local Organization Committee (i.e. Leila Amgoud, Claudette Cayrol, Véronique Debats, Marie-Christine Lagasquie-Schiex, Jérôme Men-gin, Laurent Perrussel, and Henri Prade) for all the hard work they put into making the conference a success.

March 2008

Philippe Besnard (Conference Chair)  
Sylvie Doutre (Local Organization Chair)  
Anthony Hunter (Programme Chair)

# Programme Committee

Leila Amgoud (CNRS Toulouse)	Ron Loui (Washington)
Pietro Baroni (Brescia)	Nicolas Maudet (Paris-Dauphine)
Trevor Bench-Capon (Liverpool)	Peter McBurney (Liverpool)
Gerhard Brewka (Leipzig)	Jérôme Mengin (Toulouse)
Simon Buckingham Shum (Open Univ.)	Sanjay Modgil (King's College London)
Martin Caminada (Luxembourg)	Tim Norman (Aberdeen)
Claudette Cayrol (Toulouse)	Simon Parsons (City Univ. of New York)
Carlos Chesñevar (Univ. Nacional del Sur)	Henri Prade (CNRS Toulouse)
Sylvie Coste (Artois)	Henry Prakken (Utrecht & Groningen)
Phan Minh Dung (Asian Inst. of Technologies)	Iyad Rahwan (British Univ. in Dubai)
Paul E. Dunne (Liverpool)	Chris Reed (Dundee)
John Fox (Oxford)	Guillermo Simari (Univ. Nacional del Sur)
Massimiliano Giacomin (Brescia)	Francesca Toni (Imperial College London)
Tom Gordon (Fraunhofer FOKUS)	Paolo Torroni (Bologna)
Floriana Grasso (Liverpool)	Bart Verheij (Groningen)
Antonis Kakas (Cyprus)	Gerard Vreeswijk (Utrecht)
Gabriele Kern-Isberner (Dortmund)	Doug Walton (Winnipeg)
Paul Krause (Surrey)	Michael Wooldridge (Liverpool)

# Contents

Preface <i>Philippe Besnard, Sylvie Doutre and Anthony Hunter</i>	v
Programme Committee	vi
A Level-Based Approach to Computing Warranted Arguments in Possibilistic Defeasible Logic Programming <i>Teresa Alsinet, Carlos Chesñevar and Lluís Godo</i>	1
Measures for Persuasion Dialogs: A Preliminary Investigation <i>Leila Amgoud and Florence Dupin De Saint Cyr</i>	13
Resolution-Based Argumentation Semantics <i>Pietro Baroni and Massimiliano Giacomin</i>	25
A Systematic Classification of Argumentation Frameworks Where Semantics Agree <i>Pietro Baroni and Massimiliano Giacomin</i>	37
Asking the Right Question: Forcing Commitment in Examination Dialogues <i>Trevor J.M. Bench-Capon, Sylvie Doutre and Paul E. Dunne</i>	49
Ontological Foundations for Scholarly Debate Mapping Technology <i>Neil Benn, Simon Buckingham Shum, John Domingue and Clara Mancini</i>	61
Investigating Stories in a Formal Dialogue Game <i>Floris Bex and Henry Prakken</i>	73
Modeling Persuasiveness: Change of Uncertainty Through Agents' Interactions <i>Katarzyna Budzyńska, Magdalena Kacprzak and Paweł Rembelski</i>	85
Cohere: Towards Web 2.0 Argumentation <i>Simon Buckingham Shum</i>	97
On the Issue of Contraposition of Defeasible Rules <i>Martin Caminada</i>	109
Political Engagement Through Tools for Argumentation <i>Dan Cartwright and Katie Atkinson</i>	116
A Computational Model of Argumentation in Everyday Conversation: A Problem-Centred Approach <i>Jean-Louis Dessalles</i>	128
Towards Argumentation-Based Contract Negotiation <i>Phan Minh Dung, Phan Minh Thang and Francesca Toni</i>	134
The Computational Complexity of Ideal Semantics I: Abstract Argumentation Frameworks <i>Paul E. Dunne</i>	147

Focused Search for Arguments from Propositional Knowledge <i>Vasiliki Efstathiou and Anthony Hunter</i>	159
Decision Rules and Arguments in Defeasible Decision Making <i>Edgardo Ferretti, Marcelo L. Errecalde, Alejandro J. García and Guillermo R. Simari</i>	171
Hybrid Argumentation and Its Properties <i>Dorian Gaertner and Francesca Toni</i>	183
Requirements for Reflective Argument Visualization Tools: A Case for Using Validity as a Normative Standard <i>Michael H.G. Hoffmann</i>	196
Argumentation Using Temporal Knowledge <i>Nicholas Mann and Anthony Hunter</i>	204
Strong and Weak Forms of Abstract Argument Defense <i>Diego C. Martínez, Alejandro J. García and Guillermo R. Simari</i>	216
Basic Influence Diagrams and the Liberal Stable Semantics <i>Paul-Amaury Matt and Francesca Toni</i>	228
Integrating Object and Meta-Level Value Based Argumentation <i>Sanjay Modgil and Trevor Bench-Capon</i>	240
Applying Preferences to Dialogue Graphs <i>Sanjay Modgil and Henry Prakken</i>	252
A Methodology for Action-Selection Using Value-Based Argumentation <i>Fahd Saud Nawwab, Trevor Bench-Capon and Paul E. Dunne</i>	264
Semantics for Evidence-Based Argumentation <i>Nir Oren and Timothy J. Norman</i>	276
Argument Schemes and Critical Questions for Decision Aiding Process <i>Wassila Ouerdane, Nicolas Maudet and Alexis Tsoukias</i>	285
Arguments in OWL: A Progress Report <i>Iyad Rahwan and Bita Banihashemi</i>	297
AIF <sup>+</sup> : Dialogue in the Argument Interchange Format <i>Chris Reed, Simon Wells, Joseph Devereux and Glenn Rowe</i>	311
Heuristics in Argumentation: A Game-Theoretical Investigation <i>Régis Riveret, Henry Prakken, Antonino Rotolo and Giovanni Sartor</i>	324
Argument Theory Change: Revision Upon Warrant <i>Nicolás D. Rotstein, Martín O. Moguillansky, Marcelo A. Falappa, Alejandro J. García and Guillermo R. Simari</i>	336
Diagramming the Argument Interchange Format <i>Glenn Rowe and Chris Reed</i>	348
Dungine: A Java Dung Reasoner <i>Matthew South, Gerard Vreeswijk and John Fox</i>	360

Agent Dialogue as Partial Uncertain Argumentation and Its Fixpoint Semantics <i>Takayoshi Suzuki and Hajime Sawamura</i>	369
A Distributed Argumentation Framework Using Defeasible Logic Programming <i>Matthias Thimm and Gabriele Kern-Isberner</i>	381
On the Relationship of Defeasible Argumentation and Answer Set Programming <i>Matthias Thimm and Gabriele Kern-Isberner</i>	393
Arguments from Experience: The PADUA Protocol <i>Maya Wardeh, Trevor Bench-Capon and Frans Coenen</i>	405
Modelling Judicial Context in Argumentation Frameworks <i>Adam Wyner and Trevor Bench-Capon</i>	417
Author Index	429

This page intentionally left blank

# A Level-based Approach to Computing Warranted Arguments in Possibilistic Defeasible Logic Programming

Teresa ALSINET<sup>a,1</sup>, Carlos CHESÑEVAR<sup>b</sup> and Lluís GODO<sup>c</sup>

<sup>a</sup> Department of Computer Science. University of Lleida, SPAIN

<sup>b</sup> Dept. of Computer Science & Eng. Universidad Nacional del Sur, ARGENTINA

<sup>c</sup> Artificial Intelligence Research Institute (IIA-CSIC), Bellaterra, SPAIN

**Abstract.** Possibilistic Defeasible Logic Programming (P-DeLP) is an argumentation framework based on logic programming which incorporates a treatment of possibilistic uncertainty at object-language level. In P-DeLP, the closure of justified conclusions is not always consistent, which has been detected to be an anomaly in the context of so-called rationality postulates for rule-based argumentation systems. In this paper we present a novel level-based approach to computing warranted arguments in P-DeLP which ensures the above rationality postulate. We also show that our solution presents some advantages in comparison with the use of a transposition operator applied on strict rules.

**Keywords.** Formal models of argument, Possibilistic logic, Rationality postulates for argumentation

## 1. Introduction and motivation

Possibilistic Defeasible Logic Programming (P-DeLP) [1] is an argumentation framework based on logic programming which incorporates the treatment of possibilistic uncertainty at the object-language level. Indeed, P-DeLP is an extension of Defeasible Logic Programming (DeLP) [10], a logic programming approach to argumentation which has been successfully used to solve real-world problems in several contexts such as knowledge distribution [4] and recommendations systems [9], among others. As in the case of DeLP, the P-DeLP semantics is skeptical, based on a query-driven proof procedure which computes warranted (justified) arguments. Following the terminology used in [6], P-DeLP can be seen as a member of the family of *rule-based argumentation systems*, as it is based on a logical language defined over a set of (weighed) literals and the notions of *strict* and *defeasible* rules, which are used to characterize a P-DeLP program.

Recently Caminada & Amgoud have defined several *rationality postulates* [6] which every rule-based argumentation system should satisfy. One of such postulates (called *Indirect Consistency*) involves ensuring that the closure of warranted conclusions be guar-

---

<sup>1</sup>Correspondence to: T. Alsinet. Department of Computer Science, University of Lleida. C/Jaume II, 69. Lleida, Spain. Tel.: +34 973 70 2734; Fax: +34 973 70 2702; E-mail: tracy@diei.udl.cat

anteed to be consistent. Failing to satisfy this postulate implies some anomalies and unintuitive results (e.g. the modus ponens rule cannot be applied based on justified conclusions). A number of rule-based argumentation systems are identified in which such postulate does not hold (including DeLP [10] and Prakken & Sartor [12], among others). As an alternative to solve this problem, the use of *transposed rules* is proposed to extend the representation of strict rules. For grounded semantics, the use of a transposition operator ensures that all rationality postulates to be satisfied [6, pp.294].

In this paper we present a novel level-based approach to computing warranted arguments in P-DeLP which ensures the above rationality postulate without requiring the use of transposed rules. Additionally, in contrast with DeLP and other argument-based approaches, we do not require the use of dialectical trees as underlying structures for characterizing our proof procedure. We show that our solution presents some advantages in comparison with the use of a transposition operator applied on strict rules, which might also be problematic in some cases. In particular, we show that adding transposed rules can turn a valid (consistent) P-DeLP program into an inconsistent one, disallowing further argument-based inferences on the basis of such a program.

The rest of the paper is structured as follows. Section 2 summarizes the main elements of P-DeLP. Section 3 discusses the role of the rationality postulate of indirect consistency introduced in [5], and the solution provided in terms of a transposition operator  $Cl_{tp}$ . We also show some aspects which may be problematic from this approach in P-DeLP. Section 4 presents our new level-based definitions of warrant for P-DeLP, as well as some illustrative examples. We also show that this characterization ensures that the above postulate can be now satisfied without requiring the use of transposed rules nor the computation of dialectical trees. Finally Section 5 discusses some related work and concludes.

## 2. Argumentation in P-DeLP: an overview

In order to make this paper self-contained, we will present next the main definitions that characterize the P-DeLP framework. For details the reader is referred to [1]. The language of P-DeLP is inherited from the language of logic programming, including the usual notions of atom, literal, rule and fact, but over an extended set of atoms where a new atom “ $\sim p$ ” is added for each original atom  $p$ . Therefore, a *literal* in P-DeLP is either an atom  $p$  or a (negated) atom of the form  $\sim p$ , and a *goal* is any literal.

A *weighted clause* is a pair of the form  $(\varphi, \alpha)$ , where  $\varphi$  is a rule  $Q \leftarrow P_1 \wedge \dots \wedge P_k$  or a fact  $Q$  (i.e., a rule with empty antecedent), where  $Q, P_1, \dots, P_k$  are literals, and  $\alpha \in [0, 1]$  expresses a lower bound for the necessity degree of  $\varphi$ . We distinguish between *certain* and *uncertain* clauses. A clause  $(\varphi, \alpha)$  is referred to as certain if  $\alpha = 1$  and uncertain, otherwise. A set of P-DeLP clauses  $\Gamma$  will be deemed as *contradictory*, denoted  $\Gamma \vdash \perp$ , if, for some atom  $q$ ,  $\Gamma \vdash (q, \alpha)$  and  $\Gamma \vdash (\sim q, \beta)$ , with  $\alpha > 0$  and  $\beta > 0$ , where  $\vdash$  stands for deduction by means of the following particular instance of the *generalized modus ponens rule*:

$$\frac{(Q \leftarrow P_1 \wedge \dots \wedge P_k, \alpha) \\ (P_1, \beta_1), \dots, (P_k, \beta_k)}{(Q, \min(\alpha, \beta_1, \dots, \beta_k))} \text{ [GMP]}$$

Formally, we will write  $\Gamma \vdash (Q, \alpha)$ , where  $\Gamma$  is a set of clauses,  $Q$  is a literal and  $\alpha > 0$ , when there exists a finite sequence of clauses  $C_1, \dots, C_m$  such that  $C_m = (Q, \alpha)$  and, for each  $i \in \{1, \dots, m\}$ , either  $C_i \in \Gamma$ , or  $C_i$  is obtained by applying the *GMP* rule to previous clauses in the sequence.

A P-DeLP *program*  $\mathcal{P}$  (or just program  $\mathcal{P}$ ) is a pair  $(\Pi, \Delta)$ , where  $\Pi$  is a non-contradictory finite set of certain clauses, also referred to as *strict facts* and *rules*, and  $\Delta$  is a finite set of uncertain clauses, also referred to as *defeasible facts* and *rules*. Formally, given a program  $\mathcal{P} = (\Pi, \Delta)$ , we say that a set  $\mathcal{A} \subseteq \Delta$  of uncertain clauses is an *argument* for a goal  $Q$  with necessity degree  $\alpha > 0$ , denoted  $\langle \mathcal{A}, Q, \alpha \rangle$ , iff:

1.  $\Pi \cup \mathcal{A}$  is non contradictory;
2.  $\alpha = \max\{\beta \in [0, 1] \mid \Pi \cup \mathcal{A} \vdash (Q, \beta)\}$ , i.e.  $\alpha$  is the greatest degree of deduction of  $Q$  from  $\Pi \cup \mathcal{A}$ ;
3.  $\mathcal{A}$  is minimal wrt set inclusion, i.e. there is no  $\mathcal{A}_1 \subset \mathcal{A}$  such that  $\Pi \cup \mathcal{A}_1 \vdash (Q, \alpha)$ .

Moreover, if  $\langle \mathcal{A}, Q, \alpha \rangle$  and  $\langle \mathcal{S}, R, \beta \rangle$  are two arguments wrt a program  $\mathcal{P} = (\Pi, \Delta)$ , we say that  $\langle \mathcal{S}, R, \beta \rangle$  is a *subargument* of  $\langle \mathcal{A}, Q, \alpha \rangle$ , denoted  $\langle \mathcal{S}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle$ , whenever  $\mathcal{S} \subseteq \mathcal{A}$ . Notice that the goal  $R$  may be any subgoal associated with the goal  $Q$  in the argument  $\mathcal{A}$ . From the above definition of argument, note that if  $\langle \mathcal{S}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle$  it holds that: (i)  $\beta \geq \alpha$ , and (ii) if  $\beta = \alpha$ , then  $\mathcal{S} = \mathcal{A}$  iff  $R = Q$ .

Let  $\mathcal{P}$  be a P-DeLP program, and let  $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$  and  $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$  be two arguments wrt  $\mathcal{P}$ . We say that  $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$  counterargues  $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$  iff there exists a subargument (called *disagreement subargument*)  $\langle \mathcal{S}, Q, \beta \rangle$  of  $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$  such that  $Q_1 = \sim Q$ <sup>2</sup>. Moreover, if the argument  $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$  counterargues the argument  $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$  with disagreement subargument  $\langle \mathcal{A}, Q, \beta \rangle$ , we say that  $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$  is a *proper* (respectively *blocking*) *defeater* for  $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$  when  $\alpha_1 > \beta$  (respectively  $\alpha_1 = \beta$ ).

In P-DeLP, as in other argumentation systems [8,13], argument-based inference involves a dialectical process in which arguments are compared in order to determine which beliefs or goals are ultimately accepted (or *justified* or *warranted*) on the basis of a given program and is formalized in terms of an exhaustive dialectical analysis of all possible argumentation lines rooted in a given argument. An *argumentation line* starting in an argument  $\langle \mathcal{A}_0, Q_0, \alpha_0 \rangle$  is a sequence of arguments  $\lambda = [\langle \mathcal{A}_0, Q_0, \alpha_0 \rangle, \langle \mathcal{A}_1, Q_1, \alpha_1 \rangle, \dots, \langle \mathcal{A}_n, Q_n, \alpha_n \rangle, \dots]$  such that each  $\langle \mathcal{A}_i, Q_i, \alpha_i \rangle$  is a defeater for the previous argument  $\langle \mathcal{A}_{i-1}, Q_{i-1}, \alpha_{i-1} \rangle$  in the sequence,  $i > 0$ . In order to avoid *fallacious reasoning* additional constraints are imposed, namely:

1. **Non-contradiction:** given an argumentation line  $\lambda$ , the set of arguments of the proponent (respectively opponent) should be *non-contradictory* wrt  $\mathcal{P}$ .<sup>3</sup>
2. **Progressive argumentation:** (i) every blocking defeater  $\langle \mathcal{A}_i, Q_i, \alpha_i \rangle$  in  $\lambda$  with  $i > 0$  is defeated by a proper defeater<sup>4</sup>  $\langle \mathcal{A}_{i+1}, Q_{i+1}, \alpha_{i+1} \rangle$  in  $\lambda$ ; and (ii) each argument  $\langle \mathcal{A}_i, Q_i, \alpha_i \rangle$  in  $\lambda$ , with  $i \geq 2$ , is such that  $Q_i \neq \sim Q_{i-1}$ .

<sup>2</sup>In what follows, for a given goal  $Q$ , we will write  $\sim Q$  as an abbreviation to denote “ $\sim q$ ” if  $Q = q$ , and to denote “ $q$ ” if  $Q = \sim q$ .

<sup>3</sup>A set of arguments  $S = \bigcup_{i=1}^n \{\langle \mathcal{A}_i, Q_i, \alpha_i \rangle\}$  is defined as *contradictory* wrt a program  $\mathcal{P} = (\Pi, \Delta)$  iff  $\Pi \cup \bigcup_{i=1}^n \mathcal{A}_i$  is contradictory.

<sup>4</sup>It must be noted that the last argument in an argumentation line is allowed to be a blocking defeater for the previous one.

The non-contradiction condition disallows the use of contradictory information on either side (proponent or opponent). The first condition of progressive argumentation enforces the use of a proper defeater to defeat an argument which acts as a blocking defeater, while the second condition avoids non optimal arguments in the presence of a conflict. An argumentation line satisfying the above restrictions is called *acceptable*, and can be proven to be finite. The set of all possible acceptable argumentation lines results in a structure called *dialectical tree*. Given a program  $\mathcal{P} = (\Pi, \Delta)$  and a goal  $Q$ , we say that  $Q$  is *warranted* wrt  $\mathcal{P}$  with a *maximum necessity degree*  $\alpha$  iff there exists an argument  $\langle \mathcal{A}, Q, \alpha \rangle$ , for some  $\mathcal{A} \subseteq \Delta$ , such that: i) every acceptable argumentation line starting with  $\langle \mathcal{A}, Q, \alpha \rangle$  has an odd number of arguments; and ii) there is no other argument of the form  $\langle \mathcal{A}_1, Q, \beta \rangle$ , with  $\beta > \alpha$ , satisfying the above. In the rest of the paper we will write  $\mathcal{P} \mid \sim^w \langle \mathcal{A}, Q, \alpha \rangle$  to denote this fact.

### 3. Indirect consistency as rationality postulate. Transposition of strict rules

In a recent paper Caminada and Amgoud [6] have defined a very interesting characterization of three *rationality postulates* that –according to the authors– any rule-based argumentation system should satisfy in order to avoid anomalies and unintuitive results. We will summarize next the main aspects of these postulates, and their relationship with the P-DeLP framework. Their formalization is intentionally generic, based on a *defeasible theory*  $\mathcal{T} = \langle \mathcal{S}, \mathcal{D} \rangle$ , where  $\mathcal{S}$  is a set of strict rules and  $\mathcal{D}$  is a set of defeasible rules. The notion of negation is modelled in the standard way by means of a function “ $-$ ”. An *argumentation system* is a pair  $\langle \text{Args}, \text{Def} \rangle$ , where  $\text{Args}$  is a set of arguments (based on a defeasible theory) and  $\text{Def} \subseteq \text{Args} \times \text{Args}$  is a defeat relation. The *closure* of a set of literals  $\mathcal{L}$  under the set  $\mathcal{S}$ , denoted  $CL_{\mathcal{S}}(\mathcal{L})$  is the smallest set such that  $\mathcal{L} \subseteq CL_{\mathcal{S}}(\mathcal{L})$ , and if  $\phi_1, \dots, \phi_n \rightarrow \psi \in \mathcal{S}$ , and  $\phi_1, \dots, \phi_n \in CL_{\mathcal{S}}(\mathcal{L})$ , then  $\psi \in CL_{\mathcal{S}}(\mathcal{L})$ . A set of literals  $\mathcal{L}$  is *consistent* iff there do not exist  $\psi, \phi \in \mathcal{L}$  such that  $\psi = -\phi$ , otherwise it is said to be *inconsistent*. An argumentation system  $\langle \text{Args}, \text{Def} \rangle$  can have different *extensions*  $E_1, E_2, \dots, E_n$  ( $n \geq 1$ ) according to the adopted semantics. The conclusions associated with those arguments belonging to a given extension  $E_i$  are defined as  $\text{Concs}(E_i)$ , and the *output* of the argumentation system is defined skeptically as  $\text{Output} = \bigcap_{i=1 \dots n} \text{Concs}(E_i)$ .

On the basis of the above concepts, Caminada & Amgoud [6, pp.294] present three important postulates: *direct consistency*, *indirect consistency* and *closure*. Let  $\mathcal{T}$  be a defeasible theory,  $\langle \text{Args}, \text{Def} \rangle$  an argumentation system built from  $\mathcal{T}$ ,  $\text{Output}$  the set of justified (warranted) conclusions, and  $E_1, \dots, E_n$  its extensions under a given semantics. Then these three postulates are defined as follows:

- $\langle \text{Args}, \text{Def} \rangle$  satisfies **closure** iff (1)  $\text{Concs}(E_i) = CL_{\mathcal{S}}(\text{Concs}(E_i))$  for each  $1 \leq i \leq n$  and (2)  $\text{Output} = CL_{\mathcal{S}}(\text{Output})$ .
- $\langle \text{Args}, \text{Def} \rangle$  satisfies **direct consistency** iff (1)  $\text{Concs}(E_i)$  is consistent for each  $1 \leq i \leq n$  and (2)  $\text{Output}$  is consistent.
- $\langle \text{Args}, \text{Def} \rangle$  satisfies **indirect consistency** iff (1)  $CL_{\mathcal{S}}(\text{Concs}(E_i))$  is consistent for each  $1 \leq i \leq n$  and (2)  $CL_{\mathcal{S}}(\text{Output})$  is consistent.

Closure accounts for requiring that the set of justified conclusions as well as the set of conclusions supported by each extension are closed. Direct consistency implies that

the set of justified conclusions and the different sets of conclusions corresponding to each extension are consistent. Indirect consistency involves a more subtle case, requiring that the closure of both  $\text{Concs}(E_i)$  and  $\text{Output}$  is consistent.

Caminada and Amgoud show that many rule-based argumentation system (e.g. Prakken & Sartor [12] and DeLP [10]) fail to satisfy indirect consistency, detecting as a solution the definition of a special *transposition operator*  $Cl_{tp}$  for computing the closure of strict rules. This accounts for taking every strict rule  $r = \phi_1, \phi_2, \dots, \phi_n \rightarrow \psi$  as a material implication in propositional logic which is equivalent to the disjunction  $\phi_1 \vee \phi_2 \vee \dots \vee \phi_n \vee \neg\psi$ . From that disjunction different rules of the form  $\phi_1, \dots, \phi_{i-1}, \neg\psi, \phi_{i+1}, \dots, \phi_n \rightarrow \neg\phi_i$  can be obtained (*transpositions* of  $r$ ). If  $\mathcal{S}$  is a set of strict rules,  $Cl_{tp}(\mathcal{S})$  is the minimal set such that (1)  $\mathcal{S} \subseteq Cl_{tp}(\mathcal{S})$  and (2) If  $s \in Cl_{tp}(\mathcal{S})$  and  $t$  is a transposition of  $s$ , then  $t \in Cl_{tp}(\mathcal{S})$ . The use of such an operator allows the three rationality postulates to be satisfied in the case of the grounded extension (which corresponds to the one associated with systems like DeLP or P-DeLP).

**Theorem 1** [6] *Let  $\langle \text{Args}, \text{Def} \rangle$  be an argumentation system built from  $\langle Cl_{tp}(\mathcal{S}), \mathcal{D} \rangle$ , where  $Cl_{tp}(\mathcal{S})$  is consistent,  $\text{Output}$  is the set of justified conclusions and  $E$  its grounded extension. Then  $\langle \text{Args}, \text{Def} \rangle$  satisfies closure and indirect consistency.*

Caminada & Amgoud show that DeLP does not satisfy the indirect consistency postulate. The same applies for P-DeLP, as illustrated next. Consider the program  $\mathcal{P} = (\Pi, \Delta)$ , where  $\Pi = \{ (y, 1), (\neg y \leftarrow a \wedge b, 1) \}$  and  $\Delta = \{ (a, 0.9), (b, 0.9) \}$ . It is easy to see that arguments  $\langle \{(a, 0.9)\}, a, 0.9 \rangle$  and  $\langle \{(b, 0.9)\}, b, 0.9 \rangle$  have no defeaters wrt  $\mathcal{P}$ . Thus  $\{y, a, b\} = \text{Output}$  turns out to be warranted, and it holds that  $y, \neg y \in CL_{Cl_{tp}(\Pi)}(\{y, a, b\})$ , so that indirect consistency does not hold.

We think that Caminada & Amgoud's postulate of indirect consistency is indeed valuable for rule-based argumentation systems, as in some sense it allows to perform “forward reasoning” from warranted literals. However, P-DeLP and DeLP are *Horn*-based systems, so that strict rules should be read as a kind of inference rules rather than as proper material implications. In this respect, the use of transposed rules might lead to unintuitive situations in a logic programming context. Consider e.g. the program  $\mathcal{P} = \{ (q \leftarrow p \wedge r, 1), (s \leftarrow \neg r, 1), (p, 1), (\neg q, 1), (\neg s, 1) \}$ . In P-DeLP, the facts  $(p, 1)$ ,  $(\neg q, 1)$  and  $(\neg s, 1)$  would be warranted literals. However, the closure under transposition  $Cl_{tp}(\mathcal{P})$  would include the rule  $(\neg r \leftarrow p \wedge \neg q, 1)$ , resulting in inconsistency (both  $(\neg s, 1)$  and  $(s, 1)$  can be derived), so that the whole program would be deemed as invalid. Our goal is to retain a Horn-based view for a rule-based argumentation system like P-DeLP, satisfying at the same time the indirect consistency postulate. To do this, instead of making use of transposed rules, we will introduce a new approach based on the notion of *level-based* warranted literals which is as discussed in the next Section.

## 4. A level-based approach to computing warranted arguments

In a logic programming system like P-DeLP the use of transposed rules to ensure indirect consistency may have some drawbacks that have to be taken into consideration. Apart from the problem mentioned at the end of last section of turning an apparently valid program into a non-valid one, there are two other issues: (i) a computational lim-

itation, in the sense that extending a P-DeLP program with all possible transpositions of every strict rule may lead to an important increase in the number of arguments to be computed; and (ii) when doing so, the system can possibly establish as warranted goals conclusions which are not explicitly expressed in the original program. For instance, consider the program  $\mathcal{P} = \{(\sim y \leftarrow a \wedge b, 1), (y, 1), (a, 0.9), (b, 0.7)\}$ . Transpositions of the strict rule  $(\sim y \leftarrow a \wedge b, 1)$  are  $(\sim a \leftarrow y \wedge b, 1)$  and  $(\sim b \leftarrow y \wedge a, 1)$ . Then, the argument  $\langle \mathcal{A}, \sim b, 0.9 \rangle$ , with  $\mathcal{A} = \{(y, 1), (a, 0.9), (\sim b \leftarrow a \wedge y, 1)\}$ , is warranted wrt  $\mathcal{P}$ , although no explicit information is given for the literal  $\sim b$  in  $\mathcal{P}$ . In this paper we will provide a new formal definition of warranted goal with maximum necessity degree which will take into account direct and indirect conflicts between arguments. Indirect conflicts will be detected without explicitly transposing strict rules, distinguishing between *warranted* and *blocked* goals.

Direct conflicts between arguments refer to the case of both proper and blocking defeaters. For instance, consider the program  $\mathcal{P} = \{(a \leftarrow b, 0.9), (b, 0.8), (\sim b, 0.8)\}$ . Thus arguments  $\langle \{(b, 0.8)\}, b, 0.8 \rangle$  and  $\langle \{(\sim b, 0.8)\}, \sim b, 0.8 \rangle$  are a pair of blocking defeaters expressing (direct) contradictory information, and therefore  $b$  and  $\sim b$  will be considered a pair of blocked goals with maximum necessity degree 0.8. Note that although the argument  $\langle \{(\sim b, 0.8)\}, \sim b, 0.8 \rangle$  is a blocking defeater for the argument  $\langle \mathcal{A}, a, 0.8 \rangle$ , with  $\mathcal{A} = \{(a \leftarrow b, 0.9), (b, 0.8)\}$ , goals  $a$  and  $\sim b$  do not express contradictory information, and therefore  $a$  is a neither blocked nor warranted goal.

On the other hand, we will refer to indirect conflicts between arguments when there exists an inconsistency emerging from the set of certain (strict) clauses of a program and arguments with no defeaters. For instance, consider the program  $\mathcal{P} = (\Pi, \Delta)$  with  $\Pi = \{(\sim y \leftarrow a \wedge b, 1), (y, 1), (\sim x \leftarrow c \wedge d, 1), (x, 1)\}$  and  $\Delta = \{(a, 0.7), (b, 0.7), (c, 0.7), (d, 0.6)\}$ . In standard P-DeLP [1], (i.e. without extending the program with transpositions of rules of  $\Pi$ )  $\langle \{(a, 0.7)\}, a, 0.7 \rangle$  and  $\langle \{(b, 0.7)\}, b, 0.7 \rangle$  are arguments with no defeaters and therefore their conclusions would be warranted. However, since  $\Pi \cup \{(a, 0.7), (b, 0.7)\} \vdash \perp$ , arguments  $\langle \{(a, 0.7)\}, a, 0.7 \rangle$  and  $\langle \{(b, 0.7)\}, b, 0.7 \rangle$  express (indirect) contradictory information. Moreover, as both goals are supported by arguments with the same necessity degree 0.7, none of them can be warranted nor rejected, and therefore we will refer to them as (indirect) blocked goals with maximum necessity degree 0.7. On the other hand, a similar situation appears with  $\langle \{(c, 0.7)\}, c, 0.7 \rangle$  and  $\langle \{(d, 0.6)\}, d, 0.6 \rangle$ . As before,  $\Pi \cup \{(c, 0.7), (d, 0.6)\} \vdash \perp$ , but in this case the necessity degree of goal  $c$  is greater than the necessity degree of goal  $d$ . Therefore  $c$  will be considered a warranted goal with maximum necessity degree 0.7.

Let  $ARG(\mathcal{P}) = \{\langle \mathcal{A}, Q, \alpha \rangle \mid \mathcal{A} \text{ is an argument for } Q \text{ with necessity } \alpha \text{ wrt } \mathcal{P}\}$  and let  $Concl(\mathcal{P}) = \{(Q, \alpha) \mid \langle \mathcal{A}, Q, \alpha \rangle \in ARG(\mathcal{P})\}$ . An *output* for a P-DeLP program  $\mathcal{P}$  will be a pair  $(Warr, Block)$ , where  $Warr, Block \subseteq Concl(\mathcal{P})$ , denoting respectively a set of warranted and blocked goals (together with their degrees) and fulfilling a set of conditions that will ensure a proper handling of the problem of global inconsistency discussed earlier, and that will be specified in the following definition. Since the intended construction of the sets  $Warr, Block$  is done level-wise, starting from the first level and iteratively going from one level to next level below, we introduce some useful notation. Indeed, if  $1 \geq \alpha_1 > \alpha_2 > \dots > \alpha_p > 0$  are the weights appearing in arguments from  $ARG(\mathcal{P})$ , we can stratify the sets by putting  $Warr = Warr(\alpha_1) \cup \dots \cup Warr(\alpha_p)$  and similarly  $Block = Block(\alpha_1) \cup \dots \cup Block(\alpha_p)$ , where  $Warr(\alpha_i)$  and  $Block(\alpha_i)$  are respectively the sets of the warranted and blocked goals with maximum degree  $\alpha_i$ . We

will also write  $Warr(> \alpha_i)$  to denote  $\cup_{\beta > \alpha_i} Warr(\beta)$ , and analogously for  $Block(> \alpha_i)$ . In what follows, given a program  $\mathcal{P} = (\Pi, \Delta)$  we will denote by  $rules(\Pi)$  and  $facts(\Pi)$  the set of strict rules and strict facts of  $\mathcal{P}$  respectively.

**Definition 1 (Warranted and blocked goals)** *Given a program  $\mathcal{P} = (\Pi, \Delta)$ , an output for  $\mathcal{P}$  is a pair  $(Warr, Block)$  where the sets  $Warr(\alpha_i)$  and  $Block(\alpha_i)$ , for  $i = 1 \dots p$  are required to satisfy the following constraints:*

1. An argument  $\langle \mathcal{A}, Q, \alpha_i \rangle \in ARG(\mathcal{P})$  is called acceptable if it satisfies the following three conditions:
  - (i)  $(Q, \beta) \notin Warr(> \alpha_i) \cup Block(> \alpha_i)$  and  $(\sim Q, \beta) \notin Warr(> \alpha_i) \cup Block(> \alpha_i)$ , for all  $\beta > \alpha$
  - (ii) for any subargument  $\langle \mathcal{B}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha_i \rangle$  such that  $R \neq Q$ ,  $(R, \beta) \in Warr(\beta)$
  - (iii)  $rules(\Pi) \cup Warr(> \alpha_i) \cup \{(R, \alpha_i) \mid \langle \mathcal{B}, R, \alpha_i \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha_i \rangle\} \not\vdash \perp$ .
2. For each acceptable  $\langle \mathcal{A}, Q, \alpha_i \rangle \in ARG(\mathcal{P})$ ,  $(Q, \alpha_i) \in Block(\alpha_i)$  whenever
  - (i) either there exists an acceptable  $\langle \mathcal{B}, \sim Q, \alpha_i \rangle \in ARG(\mathcal{P})$ ; or
  - (ii) there exists  $G \subseteq \{(P, \alpha_i) \mid \langle \mathcal{C}, P, \alpha_i \rangle \in ARG(\mathcal{P})\}$  is acceptable and  $\sim P \notin Block(\alpha_i)\}$  such that  $rules(\Pi) \cup Warr(> \alpha_i) \cup G \not\vdash \perp$  and  $rules(\Pi) \cup Warr(> \alpha_i) \cup G \cup \{(Q, \alpha_i)\} \vdash \perp$ ;
 otherwise,  $(Q, \alpha_i) \in Warr(\alpha_i)$ .

Actually, the intuition underlying Def. 1 is as follows: an argument  $\langle \mathcal{A}, Q, \alpha \rangle$  is either warranted or blocked whenever each subargument  $\langle \mathcal{B}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle$ , with  $Q \neq R$ , is warranted; then it is finally warranted if it induces neither direct nor indirect conflicts, otherwise it is blocked. Note that the notion of argument ensures that for each argument  $\langle \mathcal{A}, Q, \alpha \rangle \in ARG(\mathcal{P})$ , the goal  $Q$  is non-contradictory wrt the set  $\Pi$  of certain clauses of  $\mathcal{P}$ . However, it does not ensure non-contradiction wrt  $\Pi$  together with the set  $Warr(> \alpha)$  of warranted goals with degree greater than  $\alpha$  (as required by the indirect consistency postulate [6]). Therefore, for each argument  $\langle \mathcal{A}, Q, \alpha \rangle \in ARG(\mathcal{P})$  satisfying that each subgoal is warranted, the goal  $Q$  can be warranted at level  $\alpha$  only after explicitly checking indirect conflicts wrt the set  $Warr(> \alpha)$ , i.e. after verifying that  $rules(\Pi) \cup Warr(> \alpha) \cup \{(Q, \alpha)\} \not\vdash \perp$ . For instance, consider the program  $\mathcal{P} = (\Pi, \Delta)$  with

$$\begin{aligned}\Pi &= \{(y, 1), (\sim y \leftarrow a \wedge c, 1)\} \text{ and} \\ \Delta &= \{(a, 0.9), (b, 0.9), (c \leftarrow b, 0.8)\}.\end{aligned}$$

According to Def. 1, the goal  $y$  is warranted with necessity degree 1 and goals  $a$  and  $b$  are warranted with necessity degree 0.9. Then

$$A = \langle \{(b, 0.9), (c \leftarrow b, 0.8)\}, c, 0.8 \rangle$$

is an argument for  $c$  such that the subargument  $\langle \{(b, 0.9)\}, b, 0.9 \rangle$  is warranted. However, as  $rules(\Pi) \cup \{(y, 1), (a, 0.9), (b, 0.9)\} \cup \{(c, 0.8)\} \vdash \perp$ ,  $A$  is not an acceptable argument wrt  $\mathcal{P}$  and thus, the goal  $c$  is neither warranted nor blocked wrt  $\mathcal{P}$ .

Suppose now in Def. 1, that an argument  $\langle \mathcal{A}, Q, \alpha \rangle \in ARG$  involves a warranted subgoal with necessity degree  $\alpha$ . Then  $Q$  can be warranted only after explicitly checking

indirect conflicts wrt its set of subgoals, i.e. after verifying that  $\text{rules}(\Pi) \cup \text{Warr}(> \alpha) \cup \{(R, \alpha) \mid \langle \mathcal{B}, R, \alpha \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle\} \not\vdash \perp$ . For instance, consider the program  $\mathcal{P} = (\Pi, \Delta)$ , with

$$\begin{aligned}\Pi &= \{(y, 1), (\sim y \leftarrow a \wedge b, 1)\} \text{ and} \\ \Delta &= \{(a, 0.7), (b \leftarrow a, 0.7)\}.\end{aligned}$$

Then  $y$  and  $a$  are warranted goals with necessity degrees 1 and 0.7, respectively, and although it is not possible to compute a defeater for the argument

$$B = \langle \{(a, 0.7), (b \leftarrow a, 0.7)\}, b, 0.7 \rangle$$

in  $\mathcal{P}$  and the subgoal  $a$  is warranted with necessity degree 0.7,  $B$  is not an acceptable argument wrt  $\mathcal{P}$  since  $\text{rules}(\Pi) \cup \{(y, 1), (a, 0.7)\} \cup \{(b, 0.7)\} \vdash \perp$  and thus, the goal  $b$  is neither warranted nor blocked wrt  $\mathcal{P}$ . Finally, note that in Def. 1, direct conflicts invalidate possible indirect conflicts in the following sense. Consider the program  $\mathcal{P} = (\Pi, \Delta)$ , with

$$\begin{aligned}\Pi &= \{(y \leftarrow a, 1), (\sim y \leftarrow b \wedge c, 1)\} \text{ and} \\ \Delta &= \{(a, 0.7), (b, 0.7), (c, 0.7), (\sim c, 0.7)\}.\end{aligned}$$

Then,  $c$  and  $\sim c$  are blocked goals with necessity degree 0.7 and thus  $a$ ,  $b$  and  $y$  are warranted goals with necessity degree 0.7. The next example illustrates some interesting cases of the notion of warranted and blocked goals in P-DeLP.

**Example 2** Consider the program  $\mathcal{P}_1 = (\Pi_1, \Delta_1)$ , with

$$\begin{aligned}\Pi_1 &= \{(y, 1), (\sim y \leftarrow a \wedge b, 1)\} \text{ and} \\ \Delta_1 &= \{(a, 0.7), (b, 0.7), (\sim a, 0.5)\}.\end{aligned}$$

According to Def. 1,  $(y, 1)$  is warranted and  $(a, 0.7)$  and  $(b, 0.7)$  are blocked. Then, as  $a$  is blocked with necessity degree 0.7,  $\langle \{(\sim a, 0.5)\}, \sim a, 0.5 \rangle$  is not an acceptable argument and hence the goal  $\sim a$  is neither warranted nor blocked wrt  $\mathcal{P}_1$ .

Now consider the program  $\mathcal{P}_2 = (\Pi_2, \Delta_2)$  with

$$\begin{aligned}\Pi_2 &= \{(y, 1), (\sim y \leftarrow a \wedge c, 1)\} \text{ and} \\ \Delta_2 &= \{(a, 0.9), (b, 0.9), (c \leftarrow b, 0.9)\}.\end{aligned}$$

According to Def. 1,  $(y, 1)$  and  $(b, 0.9)$  are warranted. On the other hand,  $\langle \{(a, 0.9)\}, a, 0.9 \rangle$  is an argument for  $a$  with an empty set of subarguments and  $\langle \{(b, 0.9), (c \leftarrow b, 0.9)\}, c, 0.9 \rangle$  is an argument for  $c$  satisfying that the subargument  $\langle \{(b, 0.9)\}, b, 0.9 \rangle$  is warranted. However, as  $\{(\sim y \leftarrow a \wedge c, 1)\} \cup \{(y, 1), (b, 0.9)\} \cup \{(a, 0.9), (c, 0.9)\} \vdash \perp$ ,  $a$  and  $c$  are a pair of blocked goals wrt  $\mathcal{P}_2$  with necessity degree 0.9.

Finally, consider the program  $\mathcal{P}_3 = (\Pi_2, \Delta_3)$  with

$$\Delta_3 = \{(a, 0.9), (c, 0.9), (b \leftarrow c, 0.9), (d \leftarrow a \wedge c, 0.9)\}.$$

In that case  $(y, 1)$  is warranted and  $(a, 0.9)$  and  $(c, 0.9)$  are blocked. Then, according to Def. 1, as  $c$  is a blocked goal with necessity 0.9,  $\langle \{(c, 0.9), (b \leftarrow c, 0.9)\}, b, 0.9 \rangle$  is not an acceptable argument and hence the goal  $b$  is neither warranted nor blocked wrt  $\mathcal{P}_3$ . Notice that since  $a$  and  $c$  are contradictory wrt  $\Pi_2$ , no argument can be computed for goal  $d$ .

It can be shown that if  $(Warr, Block)$  is an output<sup>5</sup> of a P-DeLP program, the set  $Warr$  of warranted goals (according to Def. 1) is indeed non-contradictory and satisfies indirect consistency with respect to the set of strict rules.

**Proposition 3 (Indirect consistency)** *Let  $\mathcal{P} = (\Pi, \Delta)$  be a P-DeLP program and let  $(Warr, Block)$  be an output for  $\mathcal{P}$ . Then:*

- (i)  $facts(\Pi) \subseteq Warr$ ,
- (ii)  $Warr \not\vdash \perp$ , and
- (iii) if  $rules(\Pi) \cup Warr \vdash (Q, \alpha)$  then  $(Q, \beta) \in Warr$  for some  $\beta \geq \alpha$ .

*Proof:* We prove (ii) and (iii), as (i) is straightforward.

(ii) Suppose that for some goal  $Q$ ,  $\{(Q, \alpha), (\sim Q, \beta)\} \subseteq Warr$ . Then, there should exist  $\mathcal{A} \subseteq \Delta$  and  $\mathcal{B} \subseteq \Delta$  such that  $\Pi \cup \mathcal{A} \vdash (Q, \alpha)$  and  $\Pi \cup \mathcal{B} \vdash (\sim Q, \beta)$ . If  $\alpha = \beta$ ,  $\langle \mathcal{A}, Q, \alpha \rangle$  and  $\langle \mathcal{B}, \sim Q, \beta \rangle$  are a pair of blocking arguments; otherwise, one is a proper defeater for the other one and  $rules(\Pi) \cup \{(Q, \alpha), (\sim Q, \beta)\} \vdash \perp$ . Hence, by Def. 1,  $\{(Q, \alpha), (\sim Q, \beta)\} \not\subseteq Warr$ .

(iii) Suppose that, for some goal  $Q$ ,  $rules(\Pi) \cup Warr \vdash (Q, \alpha)$  and  $(Q, \beta) \notin Warr$ , for all  $\beta \geq \alpha$ . Then, there should exist a strict rule in  $\Pi$  of the form  $(Q \leftarrow P_1 \wedge \dots \wedge P_k, 1)$  such that either for each  $i = 1, \dots, k$ ,  $(P_i, \alpha_i) \in Warr$  or, recursively,  $rules(\Pi) \cup Warr \vdash (P_i, \alpha_i)$ , and  $\min(\alpha_1, \dots, \alpha_k) = \alpha$ . Now, if  $(Q, \alpha) \notin Warr$ , by Def. 1, it follows that either  $(Q, \beta) \in Warr$  or  $(\sim Q, \beta) \in Warr$  for some  $\beta > \alpha$ , or  $rules(\Pi) \cup Warr \vdash (\sim Q, \alpha)$ . As  $\alpha = \min(\alpha_1, \dots, \alpha_k)$ , it follows that  $\alpha = \alpha_i$ , for some  $1 \leq i \leq k$ . Then, if  $(\sim Q, \beta) \in Warr$ , with  $\beta > \alpha$ , or  $\Pi \cup Warr \vdash (\sim Q, \alpha)$ , by Def. 1, there should exist, at least, a goal  $P_i$ , with  $1 \leq i \leq k$ , such that  $(P_i, \alpha_i) \notin Warr$  and  $rules(\Pi) \cup Warr \not\vdash (P_i, \alpha_i)$ . Hence, if  $rules(\Pi) \cup Warr \vdash (Q, \alpha)$ , then  $(Q, \beta) \in Warr$ , for some  $\beta \geq \alpha$ .  $\square$

Actually, condition (iii) above can be read also as saying that  $Warr$  satisfies the *closure* postulate (somewhat softened) with respect to the set of strict rules. Indeed, it could be recovered in the full sense if the deduction characterized by  $\vdash$  would be defined taking only into account those derivations yielding maximum degrees of necessity.

Next we show that if  $(Warr, Block)$  is an output of a P-DeLP program  $\mathcal{P} = (\Pi, \Delta)$ , the set  $Warr$  of warrented goals contains indeed each literal  $Q$  satisfying that  $\mathcal{P}^* \mid \sim^w \langle \mathcal{A}, Q, \alpha \rangle$  and  $\Pi \cup \mathcal{A} \vdash (Q, \alpha)$ , with  $\mathcal{P}^* = (\Pi \cup Cl_{tp}(rules(\Pi)), \Delta)$  and whenever  $\Pi \cup Cl_{tp}(rules(\Pi))$  is non-contradictory.

**Proposition 4** *Let  $\mathcal{P} = (\Pi, \Delta)$  be a P-DeLP program such that  $\Pi \cup Cl_{tp}(rules(\Pi))$  is non-contradictory and let  $Q$  be a literal such that  $\mathcal{P}^* \mid \sim^w \langle \mathcal{A}, Q, \alpha \rangle$ . If  $\Pi \cup \mathcal{A} \vdash (Q, \alpha)$ ,  $(Q, \alpha) \in Warr$  for all output  $(Warr, Block)$  of  $\mathcal{P}$ .<sup>6</sup>*

Notice that the inverse of Prop. 4 does not hold; i.e. assuming that  $\Pi \cup Cl_{tp}(rules(\Pi))$  is non-contradictory it can be the case that  $(Q, \alpha) \in Warr$  and  $(Q, \alpha)$  is not warranted wrt the extended program  $\mathcal{P}^*$ . This is due to the fact that the new level-wise approach

---

<sup>5</sup>We remark that, as it will be discussed at the end of the section, a P-DeLP program may have multiple outputs.

<sup>6</sup>In what follows, proofs are omitted for space reasons

for computing warranted goals allows us to consider a more specific treatment of both direct and indirect conflicts between literals. In particular we have that each blocked literal invalidates all rules in which the literal occurs. For instance, consider the program  $\mathcal{P}_1 = (\Pi_1, \Delta_1)$ , with

$$\begin{aligned}\Pi_1 &= \{(y, 1), (\sim y \leftarrow a \wedge b, 1)\} \text{ and} \\ \Delta_1 &= \{(a, 0.7), (b, 0.7), (\sim b, 0.7)\}.\end{aligned}$$

According to Def. 1,  $(y, 1)$  and  $(a, 0.7)$  are warranted and  $(b, 0.7)$  and  $(\sim b, 0.7)$  are blocked. However, when considering the extended program  $\mathcal{P}_1^* = (\Pi_1 \cup Cl_{tp}(rules(\Pi_1)), \Delta_1)$  one is considering the transposed rule  $(\sim a \leftarrow y \wedge b, 1)$  and therefore,

$$\lambda_1 = [\langle \{(a, 0.7)\}, a, 0.7 \rangle, \langle \{(b, 0.7)\}, \sim a, 0.7 \rangle]$$

is an acceptable argumentation line wrt  $\mathcal{P}_1^*$  with an even number of arguments, and thus,  $(a, 0.7)$  is not warranted wrt  $\mathcal{P}_1^*$ . Another case that can be analyzed is the following one: Consider now the program  $\mathcal{P}_2 = (\Pi_1, \Delta_2)$ , with

$$\Delta_2 = \{(a, 0.7), (b \leftarrow a, 0.7), (\sim b, 0.7)\}.$$

According to Def. 1,  $(y, 1)$  and  $(a, 0.7)$  are warranted and, as indirect conflicts are not allowed,  $\langle \mathcal{B}, b, 0.7 \rangle$  with  $\mathcal{B} = \{(b \leftarrow a, 0.7), (a, 0.7)\}$  is not an acceptable argument for  $(b, 0.7)$ , and therefore  $(\sim b, 0.7)$  is warranted. However, when considering the extended program  $\mathcal{P}_2^* = (\Pi_1 \cup Cl_{tp}(rules(\Pi_1)), \Delta_2)$ ,

$$\lambda_2 = [\langle \{(\sim b, 0.7)\}, \sim b, 0.7 \rangle, \langle \mathcal{B}, b, 0.7 \rangle]$$

is an acceptable argumentation line wrt  $\mathcal{P}_2^*$  with an even number of arguments, and thus,  $(\sim b, 0.7)$  is not warranted wrt  $\mathcal{P}_2^*$ .

The following results provide an interesting characterization of the relationship between warranted and blocked goals in a P-DeLP program.

**Proposition 5** Let  $\mathcal{P} = (\Pi, \Delta)$  be a P-DeLP program and let  $(Warr, Block)$  be an output for  $\mathcal{P}$ . Then:

1. If  $(Q, \alpha) \in Warr \cup Block$ , then there exists  $\langle \mathcal{A}, Q, \alpha \rangle \in ARG(\mathcal{P})$  and, for each subargument  $\langle \mathcal{B}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle$  with  $R \neq Q$ ,  $(R, \beta) \in Warr$ .
2. If  $(Q, \alpha) \in Warr \cup Block$ , then for every argument  $\langle \mathcal{A}, Q, \beta \rangle$ , with  $\beta > \alpha$ , there exists a subargument  $\langle \mathcal{B}, R, \gamma \rangle \sqsubseteq \langle \mathcal{A}, Q, \beta \rangle$  with  $R \neq Q$ , such that  $(R, \gamma) \notin Warr$ .
3. If  $(Q, \alpha) \in Warr$ , there is no  $\beta > 0$  such that  $(Q, \beta) \in Block$  or  $(\sim Q, \beta) \in Block$
4. If  $(Q, \alpha) \notin Warr \cup Block$  for each  $\alpha > 0$ , then either  $(\sim Q, \beta) \in Block$  for some  $\beta > 0$ , or for each argument  $\langle \mathcal{A}, Q, \alpha \rangle$ , there exists a subargument  $\langle \mathcal{B}, R, \beta \rangle \sqsubseteq \langle \mathcal{A}, Q, \alpha \rangle$  with  $R \neq Q$ , such that  $(R, \beta) \notin Warr$ , or  $rules(\Pi) \cup Warr(\geq \alpha) \cup \{(Q, \alpha)\} \vdash \perp$ .

Finally, we will come to the question of whether a program  $\mathcal{P}$  always has a *unique* output  $(Warr, Block)$  according to Def. 1. In general, the answer is yes, although we

have identified some recursive situations that might lead to different outputs. For instance, consider the program

$$\mathcal{P} = \{(p, 0.9), (q, 0.9), (\sim p \leftarrow q, 0.9), (\sim q \leftarrow p, 0.9)\}.$$

Then, according to Def. 1,  $p$  is a warranted goal iff  $q$  and  $\sim q$  are a pair of blocked goals and viceversa,  $q$  is a warranted goal iff  $p$  and  $\sim p$  are a pair of blocked goals. Hence, in that case we have two possible outputs:  $(Warr_1, Block_1)$  and  $(Warr_2, Block_2)$  where

$$\begin{aligned} Warr_1 &= \{(p, 0.9)\}, Block_1 = \{(q, 0.9), (\sim q, 0.9)\} \\ Warr_2 &= \{(q, 0.9)\}, Block_2 = \{(p, 0.9), (\sim p, 0.9)\} \end{aligned}$$

In such a case, either  $p$  or  $q$  can be warranted goals (but just one of them).<sup>7</sup> Thus, although our approach is skeptical, we can get sometimes alternative extensions for warranted beliefs. A natural solution for this problem would be adopting the intersection of all possible outputs in order to define the set of those literals which are ultimately warranted. Namely, let  $\mathcal{P}$  be a P-DeLP program, and let  $output_i(\mathcal{P}) = (Warr_i, Block_i)$  denote all possible outputs for  $\mathcal{P}$ ,  $i = 1 \dots n$ . Then the *skeptical output* of  $\mathcal{P}$  could be defined as  $output_{skep}(\mathcal{P}) = (\bigcap_{i=1..n} Warr_i, \bigcap_{i=1..n} Block_i)$ . It can be shown that  $output_{skep}(\mathcal{P})$  satisfies by construction also Prop. 3 (indirect inconsistency). It remains as a future task to study the formal properties of this definition.

## 5. Related work and conclusions

We have presented a novel level-based approach to computing warranted arguments in P-DeLP. In order to do so, we have refined the notion of conflict among arguments, providing refined definitions of blocking and proper defeat. The resulting characterization allows to compute warranted goals in P-DeLP without making use of dialectical trees as underlying structures. More importantly, we have also shown that our approach ensures the satisfiability of the indirect consistency postulate proposed in [5,6], without requiring the use of transposed rules.

Assigning levels or grades to warranted knowledge has been source of research within the argumentation community in the last years, and to the best of our knowledge can be traced back to the notion of *degree of justification* addressed by John Pollock [11]. In this paper, Pollock concentrates on the “on sum” degree of justification of a conclusion in terms of the degrees of justification of all relevant premises and the strengths of all relevant reasons. However, his work is more focused on epistemological issues than ours, not addressing the problem of indirect inconsistency, nor using the combination of logic programming and probabilistic logic to model argumentative inference. An alternative direction is explored by Besnard & Hunter [3] by characterizing *aggregation functions* such as categorisers and accumulators which allow to define more evolved forms of computing warrant (e.g. counting arguments for and against, etc.). However, this research does not address the problem of indirect consistency, and performs the grading on top of a classical first-order language, where clauses are weighed as in our case. More recently,

---

<sup>7</sup>A complete characterization of these pathological situations is a matter of current research.

the research work of Cayrol & Lagasquie-Schiex [7] pursues a more ambitious goal, providing a general framework for formalizing the notion of *graduality* in valuation models for argumentation frameworks, focusing on the valuation of arguments and the acceptability according to different semantics. However, the problem of indirect consistency is not addressed there either, and the underlying system is Dung's abstract argumentation systems, rather than a logic programming framework as in our case.

We contend that our level-based characterization of warrant can be extended to other alternative argumentation frameworks in which weighted clauses are used for knowledge representation. Part of our current research is focused on finding a suitable generalization for capturing the results presented in this paper beyond the P-DeLP framework. Actually, still in P-DeLP, instead of (numerically) weighted clauses one could equivalently consider as programs just a stratified set of clauses expressing in a qualitative form the comparative belief strength of each clause (cf. [2]).

**Acknowledgments** Authors are thankful to the anonymous reviewers for their helpful comments. This research was partially supported by CICYT Projects MULOG2 (TIN2007-68005-C04-01/04) and IEA (TIN2006-15662-C02-01/02), by CONICET (Argentina), and by the Secretaría General de Ciencia y Tecnología de la Universidad Nacional del Sur (Project PGI 24/ZN10).

## References

- [1] T. Alsinet, C. I. Chesñevar, L. Godo, and G. Simari. A logic programming framework for possibilistic argumentation: Formalization and logical properties. *Fuzzy Sets and Systems*, 2008 (to appear). Preliminary manuscript available from [http://cs.uns.edu.ar/~cic/2007/2007\\_fss.pdf](http://cs.uns.edu.ar/~cic/2007/2007_fss.pdf).
- [2] S. Benferhat, D. Dubois, and H. Prade. Some syntactic approaches to the handling of inconsistent knowledge bases: A comparative study. part ii: The prioritized case. In Ewa Orlowska, editor, *Logic at work*, volume 24, pages 473–511. Physica-Verlag , Heidelberg, 1998.
- [3] P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artif. Intell.*, 128(1-2):203–235, 2001.
- [4] R. Brena, J. Aguirre, C. Chesñevar, E. Ramírez, and L. Garrido. Knowledge and information distribution leveraged by intelligent agents. *Knowl. Inf. Syst.*, 12(2):203–227, 2007.
- [5] M. Caminada and L. Amgoud. An axiomatic account of formal argumentation. In *Proc. of the 20th AAAI Conference, Pittsburgh, Pennsylvania, USA*, pages 608–613. AAAI Press / The MIT Press, 2005.
- [6] M. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artif. Intell.*, 171(5-6):286–310, 2007.
- [7] C. Cayrol and M. Lagasquie-Schiex. Graduality in argumentation. *J. Artif. Intell. Res. (JAIR)*, 23:245–297, 2005.
- [8] C. Chesñevar, A. Maguitman, and R. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.
- [9] C. Chesñevar, A. Maguitman, and G. Simari. Argument-Based Critics and Recommenders: A Qualitative Perspective on User Support Systems. *Journal of Data and Knowledge Engineering*, 59(2):293–319, 2006.
- [10] A. García and G. Simari. Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.
- [11] J. Pollock. Defeasible reasoning with variable degrees of justification. *Artif. Intell.*, 133(1-2):233–282, 2001.
- [12] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics*, 7:25–75, 1997.
- [13] H. Prakken and G. Vreeswijk. Logical Systems for Defeasible Argumentation. In D. Gabbay and F. Guenther, editors, *Handbook of Phil. Logic*, pages 219–318. Kluwer, 2002.

# Measures for persuasion dialogs: A preliminary investigation

Leila AMGOUD<sup>a,1</sup>, Florence DUPIN DE SAINT CYR<sup>a</sup>

<sup>a</sup> Institut de Recherche en Informatique de Toulouse (IRIT), France

**Abstract.** Persuasion is one of the main types of dialogs encountered in everyday life. The basic idea behind a persuasion is that two (or more) agents disagree on a state of affairs, and each one tries to persuade the other to change his mind. For that purpose, agents exchange arguments of different strengths.

Several systems, grounded on argumentation theory, have been proposed in the literature for modeling persuasion dialogs. These systems have studied more or less deeply the different protocols required for this kind of dialogs, and have investigated different termination criteria. However, nothing is said about the *properties* of the generated dialogs, nor on the behavior of the interacting agents. Besides, analyzing dialogs is a usual task in everyday life. For instance, political debates are generally deeply dissected.

In this paper we define *measures* for analyzing dialogs from the point of view of an external agent. In particular, three kinds of measures are proposed: i) measures of the quality of the exchanged arguments in terms of their strengths, ii) measures of the behavior of each participating agent in terms of its *coherence*, its *aggressiveness* in the dialog, and finally in terms of the *novelty* of its arguments, iii) measures of the quality of the dialog itself in terms of the *relevance* and *usefulness* of its moves.

**Keywords.** Argumentation, Persuasion dialogs, Quality measures

## 1. Introduction

Since the seminal work by Walton and Krabbe [11] on the role of argumentation in dialog, and on the classification of different types of dialogs, there is an increasing interest on modeling those dialog types using argumentation techniques. Indeed, several *dialog systems* have been proposed in the literature for modeling *information seeking* dialogs (e.g. [8]), *inquiry* dialogs (e.g. [4]), *negotiation* (e.g. [10]), and finally *persuasion* dialogs (e.g. [3,6,9,13]). Persuasion dialogs are initiated from a position of conflict in which one agent believes  $p$  and the other believes  $\neg p$ , and both try to persuade the other to change its mind by presenting arguments in support of their thesis.

It is worth noticing that in all these disparate works, a dialog system is built around three main components: i) a *communication language* specifying the locutions that will be used by agents during a dialog for exchanging information, arguments, offers, etc., ii) a *protocol* specifying the set of rules governing the well-definition of dialogs, and iii)

---

<sup>1</sup>Corresponding Author: IRIT-CNRS, 118, route de Narbonne, 31062, Toulouse Cedex, France; E-mail: amgoud@irit.fr.

agents' strategies which are the different tactics used by agents for selecting their moves at each step in a dialog.

All the above systems allow agents to engage in dialogs that obey of course to the rules of the protocol. Thus, the only properties that are guaranteed for a generated dialog are those related to the protocol. For instance, one can show that a dialog terminates, the turn shifts equally between agents in that dialog (if such rule is specified by the protocol), agents can backtrack to an early move in the dialog, etc. Note that the properties inherited from a protocol concern the way the dialog is generated. However, they don't say anything about the *properties* of that dialog.

Judging the properties of a dialog may be seen as a subjective issue. Two people listening to the same political debate may disagree, for instance, on the "winner" of the debate, and more generally on their feeling about the dialog itself. Nevertheless, it is possible to define more objective criteria, for instance, the aggressiveness of each participant, the way agents may borrow ideas from each others, the self-contradiction of agents, the relevance of the exchanged information, etc.

Focusing only on persuasion dialogs, in this paper, we are concerned by analyzing already generated dialogs whatever the protocol used is and whatever the strategies of the agents are. We place ourselves in the role of an external observer that tries to evaluate the dialog. For this purpose, three kinds of measures are proposed: 1) Measures of the quality of the exchanged arguments in terms of their *weights*, 2) Measures of the behavior of each participating agent in terms of its *coherence*, its *aggressiveness* in the dialog, and finally in terms of the *source* of its arguments, 3) Measures of the properties of the dialog itself in terms of the relevance and usefulness of its moves. These measures are of great importance since they can be used as guidelines for a protocol in order to generate the "best" dialogs. They can also serve as a basis for analyzing dialogs that hold between agents.

The rest of the paper is organized as follows: Section 2 recalls the basics of argumentation theory. In Section 3, we present the basic concepts of a persuasion dialog. Section 4 details our dialog measures as well as their properties. Section 5 is devoted to some concluding remarks and conclusions.

## 2. Basics of argumentation systems

Argumentation is a reasoning model based on the construction and the comparison of arguments whose definition will be given in Section 3. In [5], an argumentation system is defined as follows:

**Definition 1 (Argumentation system)** *An argumentation system (AS) is a pair  $T = \langle \mathcal{A}, \mathcal{R} \rangle$ , where  $\mathcal{A}$  is a set of arguments and  $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$  is an attack relation. We say that an argument  $\alpha_i$  attacks an argument  $\alpha_j$  iff  $(\alpha_i, \alpha_j) \in \mathcal{R}$  (or  $\alpha_i \mathcal{R} \alpha_j$ ).*

Note that to each argumentation system is associated an oriented graph whose nodes are the different arguments, and the edges represent the attack relation between them. Let  $\mathcal{G}_T$  denote the graph associated to the argumentation system  $T = \langle \mathcal{A}, \mathcal{R} \rangle$ .

Since arguments are conflicting, it is important to know which arguments are acceptable. For that purpose, in [5], different acceptability semantics have been proposed. Let us recall them here.

**Definition 2 (Conflict-free, Defence)** *Let  $\mathcal{B} \subseteq \mathcal{A}$ .*

- $\mathcal{B}$  is conflict-free iff  $\nexists \alpha_i, \alpha_j \in \mathcal{B}$  such that  $(\alpha_i, \alpha_j) \in \mathcal{R}$ .
- $\mathcal{B}$  defends an argument  $\alpha_i$  iff for each argument  $\alpha_j \in \mathcal{A}$ , if  $(\alpha_j, \alpha_i) \in \mathcal{R}$ , then  $\exists \alpha_k \in \mathcal{B}$  such that  $(\alpha_k, \alpha_j) \in \mathcal{R}$ .

**Definition 3 (Acceptability semantics)** *Let  $\mathcal{B}$  be a conflict-free set of arguments of  $\mathcal{A}$ .*

- $\mathcal{B}$  is an admissible extension iff  $\mathcal{B}$  defends all its elements;
- $\mathcal{B}$  is a preferred extension iff it is a maximal (w.r.t. set- $\subseteq$ ) admissible extension;
- $\mathcal{B}$  is a stable extension iff it is a preferred extension that attacks w.r.t. the relation  $\mathcal{R}$  all arguments in  $\mathcal{A} \setminus \mathcal{B}$ .

Let  $\mathcal{E}_1, \dots, \mathcal{E}_n$  denote the possible extensions under a given semantics.

Now that the acceptability semantics are defined, we can define the status of any argument. As we will see, an argument may have one among three possible statuses: *skeptically accepted*, *credulously accepted* and *rejected*.

**Definition 4 (Argument status)** *Let  $\langle \mathcal{A}, \mathcal{R} \rangle$  be an argumentation system, and  $\mathcal{E}_1, \dots, \mathcal{E}_n$  its extensions under stable (resp. preferred) semantics. Let  $\alpha \in \mathcal{A}$ .*

- $\alpha$  is skeptically accepted iff  $\alpha \in \mathcal{E}_i, \forall \mathcal{E}_i \neq \emptyset$  with  $i = 1, \dots, n$ ;
- $\alpha$  is credulously accepted iff  $\alpha$  is in some extensions and not in others.
- $\alpha$  is rejected iff  $\nexists \mathcal{E}_j$  such that  $\alpha \in \mathcal{E}_j$ .

### 3. Persuasion dialogs

Let  $\mathcal{L}$  be a logical language from which arguments may be built. In our application, arguments are reasons of believing something. Throughout the paper, the structure and the origin of such arguments are supposed to be unknown. However, an argument is assumed to have at least two parts: a *support* (representing the set of premises or formulas used to build the argument) and a *conclusion* (representing the belief one wants to justify through the argument). Arguments will be denoted by lowercase Greek letters.

**Definition 5 (Argument)** *An argument  $\alpha$  is a pair  $\alpha = \langle H, h \rangle$  where  $h \in \mathcal{L}$  and  $H \subseteq \mathcal{L}$ .  $H$  is the support of the argument returned by the function  $H = \text{Support}(\alpha)$ , and  $h$  is its conclusion returned by the function  $h = \text{Conc}(\alpha)$ .*

In what follows,  $\text{arg}$  denotes a function that returns for a given set  $S \subseteq \mathcal{L}$  all the arguments that may be built from formulas of  $S$ . Thus,  $\text{arg}(\mathcal{L})$  is the set of arguments that may be built from the whole logical language  $\mathcal{L}$ .

As well established in the literature and already said, arguments may be conflicting since, for instance, they may support contradictory conclusions. In what follows,  $\mathcal{R}_{\mathcal{L}}$  is a binary relation that captures all the conflicts that may exist among arguments of  $\text{arg}(\mathcal{L})$ .

Thus,  $\mathcal{R}_{\mathcal{L}} \subseteq \arg(\mathcal{L}) \times \arg(\mathcal{L})$ . For two arguments  $\alpha, \beta \in \arg(\mathcal{L})$ , the pair  $(\alpha, \beta) \in \mathcal{R}_{\mathcal{L}}$  means that the argument  $\alpha$  attacks the argument  $\beta$ .

Let  $\text{Ag} = \{a_1, \dots, a_n\}$  be a set of symbols representing agents that may be involved in a persuasion dialog. Each agent is supposed to be able to recognize each argument of  $\arg(\mathcal{L})$  and each conflict in  $\mathcal{R}_{\mathcal{L}}$ . Note that this does not mean at all that an agent is aware of all the arguments. This assumption means that agents use the same logical language and the same definition of arguments.

A persuasion dialog consists mainly of an exchange of arguments. Of course other kinds of moves can be exchanged like questions and assertions. However, arguments play the key role in determining the outcome of the dialog. Thus, throughout the paper, we are only interested by the arguments exchanged in a dialog. The subject of such a dialog is an argument, and its aim is to compute the status of that argument. If at the end of the dialog, the argument is “skeptically accepted” or “rejected”, then we say that the dialog has *succeeded*, otherwise the dialog has *failed*.

**Definition 6 (Moves)** A move  $m \in \mathcal{M}$  is a triple  $\langle S, H, x \rangle$  such that:

- $S \in \text{Ag}$  is the agent that utters the move,  $\text{Speaker}(m) = S$
- $H \subseteq \text{Ag}$  is the set of agents to which the move is addressed,  $\text{Hearer}(m) = H$
- $x \in \arg(\mathcal{L})$  is the content of the move,  $\text{Content}(m) = x$ .

During a dialog several moves may be uttered. Those moves constitute a sequence denoted by  $\langle m_0, \dots, m_n \rangle$ , where  $m_0$  is the initial move whereas  $m_n$  is the final one. The empty sequence is denoted by  $\langle \rangle$ . For any integer  $n$ , the set of sequences of length  $n$  is denoted by  $\mathcal{M}^n$ . These sequences are built under a given protocol. A protocol amounts to define a function that associates to each sequence of moves, a set of valid moves. Several protocols have been proposed in the literature, like for instance [3,9]. In what follows, we don't focus on particular protocols.

**Definition 7 (Persuasion dialog)** A persuasion dialog  $D$  is a non-empty and finite sequence of moves  $\langle m_0, \dots, m_n \rangle$ .

The subject of  $D$  is  $\text{Subject}(D) = \text{Content}(m_0)$ , and the length of  $D$ , denoted  $|D|$ , is the number of moves  $n+1$ . Each sub-sequence  $\langle m_0, \dots, m_i \rangle$  (with  $i < n$ ) is a sub-dialog  $D^i$  of  $D$ . We will write also  $D^i \sqsubset D$ .

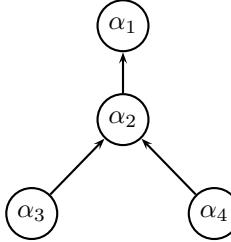
It is worth noticing that to each persuasion dialog  $D$ , one may associate an argumentation system that will be used to evaluate the status of each argument uttered during the dialog. This argumentation system is also used to compute the output of the dialog.

**Definition 8 (AS of a persuasion dialog)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog. The argumentation system of  $D$  is the pair  $\text{AS}_D = \langle \text{Args}(D), \text{Conf}(D) \rangle$  s.t.:

1.  $\text{Args}(D) = \{\text{Content}(m_i) \mid i = 0, \dots, n\}$
2.  $\text{Conf}(D) = \{(\alpha, \beta) \text{ such that } \alpha, \beta \in \text{Args}(D) \text{ and } (\alpha, \beta) \in \mathcal{R}_{\mathcal{L}}\}$

In other words,  $\text{Args}(D)$  and  $\text{Conf}(D)$  return respectively, the set of arguments exchanged during the dialog and the different conflicts among those arguments.

**Example 1** Let  $D$  be the following persuasion dialog between two agents  $a_1$  and  $a_2$ .  $D = \langle \langle a_1, \{a_2\}, \alpha_1 \rangle, \langle a_2, \{a_1\}, \alpha_2 \rangle, \langle a_1, \{a_2\}, \alpha_3 \rangle, \langle a_1, \{a_2\}, \alpha_4 \rangle, \langle a_2, \{a_1\}, \alpha_1 \rangle \rangle$ . Let us assume that there exist conflicts in  $\mathcal{R}_C$  among some of these arguments. Those conflicts are summarized in the figure below.



In this case,  $\text{Args}(D) = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$  and  $\text{Conf}s(D) = \{(\alpha_2, \alpha_1), (\alpha_3, \alpha_2), (\alpha_4, \alpha_2)\}$ .

**Property 1** Let  $D$  be a persuasion dialog.  $\forall D^j$  such that  $D^j \sqsubset D$ ,  $\text{Args}(D^j) \subseteq \text{Args}(D)$ , and  $\text{Conf}s(D^j) \subseteq \text{Conf}s(D)$ .

Any dialog has an output. In case of a persuasion, the output of a dialog is either the status of the argument under discussion (i.e. the subject) when that status is “skeptically accepted” or “rejected”, or failure in case the status of the subject is “credulously accepted”. The idea is that a dialog succeeds as soon as the status of the subject is determined, and thus a winner agent is known. However, when an argument is credulously accepted, this means that each agent keeps its position w.r.t. the subject, and the dialog fails to meet its objective.

**Definition 9 (Output of a persuasion dialog)** Let  $D$  be a persuasion dialog. The output of  $D$ , denoted by  $\text{Output}(D)$  is

$$\begin{cases} A & \text{iff } \text{Subject}(D) \text{ is skeptically accepted in } \text{AS}_D \\ R & \text{iff } \text{Subject}(D) \text{ is rejected in } \text{AS}_D \\ \text{Fail} & \text{iff } \text{Subject}(D) \text{ is credulously accepted in } \text{AS}_D \end{cases}$$

It may be the case that from the set of formulas involved in a set  $E$  of arguments, it is possible to build new arguments that do not belong to  $E$ . Let

$$\text{Formulas}(E) = \cup_{\alpha \in E} \text{Support}(\alpha)$$

be that set of formulas. Due to the monotonic construction of arguments,  $E \subseteq \text{arg}(\text{Formulas}(E))$  but the reverse is not necessarily true. Indeed, an argument remains always an argument even when new attackers are received. However, its status is non-monotonic, and may change. As a consequence of the previous inclusion, new conflicts may also appear among those new arguments, and even between new arguments and elements of  $E$ . This shows clearly that the argumentation system associated with a dialog is not necessarily “complete”. In what follows, we define the complete version of an argumentation system associated with a given dialog  $D$ .

**Definition 10 (Complete AS)** Let  $D$  be a persuasion dialog, and  $\text{AS}_D = \langle \text{Args}(D), \text{Conf}s(D) \rangle$  its associated AS.

The complete AS is  $\text{CAS}_D = \langle \text{arg}(\text{Formulas}(\text{Args}(D))), \mathcal{R}_c \rangle$  where  $\mathcal{R}_c = \{(\alpha, \beta) \text{ s.t. } \alpha, \beta \in \text{arg}(\text{Formulas}(\text{Args}(D))) \text{ and } (\alpha, \beta) \in \mathcal{R}_C\}$ .

Recall that  $\text{Args}(D) \subseteq \arg(\text{Formulas}(\text{Args}(D)))$  and  $\text{Conf}s(D) \subseteq \mathcal{R}_c \subseteq \mathcal{R}_{\mathcal{L}}$ . Note that the status of an argument  $\alpha$  in the system  $\text{AS}_D$  is not necessarily the same as in the complete system  $\text{CAS}_D$ .

## 4. Measuring persuasion dialogs

In this section we discuss different measures of persuasion dialogs. Three aspects can be analyzed: 1) the quality of the exchanged arguments during a persuasion dialog, 2) the agent's behavior, and 3) the properties of the dialog itself.

### 4.1. Measuring the quality of arguments

During a dialog, agents utter arguments that may have different *weights*. A weight may highlight the quality of information involved in the argument in terms for instance of its certainty degree. It may also be related to the cost of revealing that information. In [1], several definitions of such arguments' weights have been proposed, and their use for comparing arguments has been made explicit. It is worth noticing that the same argument may not have the same weight from one agent to another. In what follows, a weight in terms of a numerical value is associated to each argument. The greater this value is, the better the argument.

$$\text{weight} : \arg(\mathcal{L}) \longrightarrow \mathbb{N}^*$$

The function `weight` is given by the agent which wants to analyze the dialog. Thus, it may be given by an agent that is involved in the dialog, or by an external one. On the basis of arguments' weights, it is possible to compute the weight of a dialog as follows:

**Definition 11 (Measure of dialog weight)** *Let  $D$  be a persuasion dialog. The weight of  $D$  is  $\text{Weight}(D) = \sum_{\alpha_i \in \text{Args}(D)} \text{weight}(\alpha_i)$*

It is clear that this measure is monotonic. Formally:

**Property 2** *Let  $D$  be a persuasion dialog.  $\forall D^j \sqsubset D$  then  $\text{Weight}(D^j) \leq \text{Weight}(D)$ .*

This measure allows to compare pairs of persuasion dialogs only on the basis of the exchanged arguments. It is even more relevant when the two dialogs have the same subject and got the same output.

It is also possible to compute the weight of arguments uttered by each agent in a given dialog. For that purpose, one needs to know what has been said by each agent. This can be computed by a simple projection on the dialog given that agent.

**Definition 12 (Dialog projection)** *Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog, and  $a_i \in \text{Ag}$ . The projection of  $D$  on agent  $a_i$  is  $D^{a_i} = \langle m_{i_1}, \dots, m_{i_k} \rangle$  such that  $0 \leq i_1 \leq \dots \leq i_k \leq n$  and  $\forall l \in [1, k], m_{i_l} \in D$  and  $\text{Speaker}(m_{i_l}) = a_i$ .*

The contribution of each agent is defined as follows:

**Definition 13 (Measure of agent's contribution)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog, and  $a_i \in \text{Ag}$ . The contribution of agent  $a_i$  in  $D$  is

$$\text{Contr}(a_i, D) = \frac{\sum_{\alpha_i \in \text{Args}(D)} \text{weight}(\alpha_i)}{\text{Weight}(D)}.$$

**Example 2** Let us consider the persuasion dialog  $D$  presented in Example 1. Recall that  $\text{Args}(D) = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ ,  $D^{a_1} = \{\alpha_1, \alpha_3, \alpha_4\}$  and  $D^{a_2} = \{\alpha_1, \alpha_2\}$ . Suppose now that an external agent wants to analyze this dialog. The function "weight" of this agent is as follow:  $\text{weight}(\alpha_1) = 1$ ,  $\text{weight}(\alpha_2) = 4$ ,  $\text{weight}(\alpha_3) = 2$  and  $\text{weight}(\alpha_4) = 3$ . It is then clear from the definitions that the overall weight of the dialog is  $\text{Weight}(D) = 10$ . The contributions of the two agents are respectively  $\text{Contr}(a_1, D) = 6/10$  and  $\text{Contr}(a_2, D) = 5/10$ .

Consider now an example in which an agent sends several times the same argument.

**Example 3** Let us consider a persuasion dialog  $D$  such that  $\text{Args}(D) = \{\alpha, \beta\}$ .  $D^{a_1} = \{\alpha\}$  and  $D^{a_2} = \{\beta\}$ . Let us assume that there are 50 moves in this dialog of which 49 moves are uttered by agent  $a_1$  and one move uttered by agent  $a_2$ . Suppose now that an external agent wants to analyze this dialog. The function "weight" of this agent is as follow:  $\text{weight}(\alpha) = 1$  and  $\text{weight}(\beta) = 30$ . The overall weight of the dialog is  $\text{Weight}(D) = 31$ . The contributions of the two agents are respectively  $\text{Contr}(a_1, D) = 1/31$  and  $\text{Contr}(a_2, D) = 30/31$ .

It is easy to check that when the protocol under which a dialog is generated does not allow an agent to repeat an argument already given by another agent, then the sum of the contributions of the different agents is equal to 1.

**Proposition 1** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog and  $a_1, \dots, a_n$  the agents involved in it.  $\sum_{i=1, \dots, n} \text{Contr}(a_i, D) = 1$  iff  $\nexists m_i, m_j$  with  $0 \leq i, j \leq n$ , such that  $\text{Speaker}(m_i) \neq \text{Speaker}(m_j)$ , and  $\text{Content}(m_i) = \text{Content}(m_j)$ .

It is worth noticing that the measure  $\text{Contr}$  is not monotonic. Indeed, the contribution of an agent may change during the dialog. It may increase then decreases in the dialog. However, at a given step of a dialog, the contribution of the agent that will present the next move will for sure increase, whereas the contributions of the other agents may decrease. Formally:

**Proposition 2** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog and  $a_i \in \text{Ag}$ . Let  $m \in \mathcal{M}$  such that  $\text{Speaker}(m) = a_i$ . Then,  $\text{Contr}(a_i, D \oplus m) \geq \text{Contr}(a_i, D)$ , and  $\forall a_j \neq a_i$ ,  $\text{Contr}(a_j, D \oplus m) \leq \text{Contr}(a_j, D)$ , with  $D \oplus m = \langle m_0, \dots, m_n, m \rangle$ .

#### 4.2. Analyzing the behavior of agents

The behavior of an agent in a given persuasion dialog may be analyzed on the basis of three main criteria: i) its degree of *aggressiveness* in the dialog, ii) the source of its arguments, i.e. whether it builds arguments using its own formulas, or rather the ones revealed by other agents, and finally iii) its degree of *coherence* in the dialog.

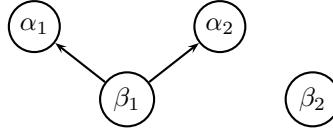
The first criterion, i.e. the aggressiveness of an agent in a dialog, amounts to compute to what extent an agent was attacking arguments sent by other agents. Such agents prefer to

destroy arguments presented by other parties rather than presenting arguments supporting their own point of view. Formally, the *aggressiveness degree* of an agent  $a_i$  towards an agent  $a_j$  during a persuasion dialog is equal to the number of its arguments that attack the other agent's arguments over the number of arguments it has uttered in that dialog.

**Definition 14 (Measure of aggressiveness)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog, and  $a_i, a_j \in \text{Ag}$ . The aggressiveness degree of agent  $a_i$  towards  $a_j$  in  $D$  is

$$\text{Agr}(a_i, a_j, D) = \frac{|\{\alpha \in \text{Args}(D^{a_i}) \text{ s.t. } \exists \beta \in \text{Args}(D^{a_j}) \text{ and } (\alpha, \beta) \in \text{Confs}(D)\}|}{|\text{Args}(D^{a_i})|}^2.$$

**Example 4** Let  $D$  be a persuasion dialog between two agents  $a_1$  and  $a_2$ . Let us assume that  $\text{Args}(D) = \{\alpha_1, \alpha_2, \beta_1, \beta_2\}$ ,  $D^{a_1} = \{\alpha_1, \alpha_2\}$  and  $D^{a_2} = \{\beta_1, \beta_2\}$ . The conflicts among the four arguments are depicted in the figure below.



The aggressiveness degrees of the two agents are respectively  $\text{Agr}(a_1, a_2, D) = 0/2 = 0$ , and  $\text{Agr}(a_2, a_1, D) = 1/2$ .

The aggressiveness degree of an agent changes as soon as a new argument is uttered by that agent. It decreases when that argument does not attack any argument of the other agent, and increases otherwise. Formally:

**Proposition 3** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog and  $a_i, a_j \in \text{Ag}$ . Let  $m \in \mathcal{M}$  such that  $\text{Speaker}(m) = a_i$ , and let  $D \oplus m = \langle m_0, \dots, m_n, m \rangle$ .  
 $\text{Agr}(a_i, a_j, D \oplus m) \geq \text{Agr}(a_i, a_j, D)$  iff  $\exists \alpha \in \text{Args}(D^{a_j})$  such that  $(\text{Content}(m), \alpha) \in \mathcal{R}_L$ , and  $\text{Agr}(a_i, a_j, D \oplus m) < \text{Agr}(a_i, a_j, D)$  otherwise.

The second criterion concerns the source of arguments. An agent can build its arguments either from its own knowledge base, thus using its own formulas, or using formulas revealed by other agents in the dialog. In [2], this idea of borrowing formulas from other agents has been presented as one of the tactics used by agents for selecting the argument to utter at a given step of a dialog. The authors argue that by doing so, an agent may minimize the risk of being attacked subsequently.

Let us now check to what extent an agent borrows information from other agents. Before that, let us first determine which formulas are owned by each agent according to what has been said in a dialog. Informally, a formula is owned by an agent, if this formula is revealed for the first time by that agent. Note that a formula revealed for the first time by agent  $a_i$  may also pertain to the base of another agent  $a_j$ . Here, we are interested by who reveals first that formula.

**Definition 15 (Agent's formulas)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog, and  $a_i \in \text{Ag}$ . The formulas owned by agent  $a_i$  are:  $\text{OwnF}(a_i, D) = \{x \in \mathcal{L} \text{ such that } \exists m_j \text{ with } \text{Speaker}(m_j) = a_i \text{ and } x \in \text{Support}(\text{Content}(m_j)) \text{ and } \nexists m_k \text{ s.t. } k < j \text{ and } \text{Speaker}(m_k) \neq a_i \text{ and } x \in \text{Support}(\text{Content}(m_k))\}$ .

---

<sup>2</sup>The expression  $|E|$  denotes the cardinal of the set E.

Now that we know which formulas are owned by each agent, we can compute the *degree of loan* for each participating agent. It may be tactically useful to turn an agents' arguments against him since they should be immune from challenge. This loan degree can thus help for evaluating the strategical behavior of an agent.

**Definition 16 (Measure of loan)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog, and  $a_i, a_j \in \text{Ag}$ . The loan degree of agent  $a_i$  from agent  $a_j$  in  $D$  is:

$$\text{Loan}(a_i, a_j, D) = \frac{|\text{Formulas}(\text{Args}(D^{a_i})) \cap \text{OwnF}(D, a_j)|}{|\text{Formulas}(\text{Args}(D^{a_i}))|}$$

The third criterion concerns the coherence of an agent. Indeed, in a persuasion dialog where an agent  $a_i$  defends its point of view, it is important that this agent does not contradict itself. In fact, there are two kinds of self contradiction:

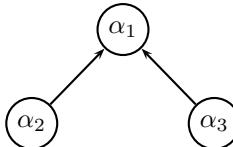
1. an *explicit* contradiction in which an agent presents at a given step of a dialog an argument, and later it attacks that argument. Such conflicts already appear in the argumentation system  $\text{AS}_{D^{a_i}} = \langle \text{Args}(D^{a_i}), \text{Confs}(D^{a_i}) \rangle$  associated to the moves uttered by agent  $a_i$ . In other words, the set  $\text{Confs}(D^{a_i})$  is not empty.
2. an *implicit* contradiction that appears only in the complete version of that system, i.e. in  $\text{CAS}_{D^{a_i}}$ .

In what follows, we will define a measure, a *degree of incoherence*, for evaluating to what extent an agent was incoherent in a dialog.

**Definition 17 (Measure of incoherence)** Let  $D$  be a persuasion dialog,  $a_i \in \text{Ag}$ , and  $\text{CAS}_{D^{a_i}} = \langle \mathcal{A}_c^{a_i}, \mathcal{R}_c^{a_i} \rangle$  its complete system. The incoherence degree of agent  $a_i$  in  $D$  is

$$\text{Inc}(a_i, D) = \frac{|\mathcal{R}_c^{a_i}|}{|\mathcal{A}_c^{a_i} \times \mathcal{A}_c^{a_i}|}$$

**Example 5** Let  $D$  be a persuasion dialog in which agent  $a_1$  has uttered two arguments  $\alpha_1$  and  $\alpha_2$ . Let us assume that from the formulas of those arguments a third argument, say  $\alpha_3$ , is built. The figure below depicts the conflicts among the three arguments. The incoherence degree of agent  $a_1$  is equal to  $2/9$ .



Note that, the above definition is general enough to capture both explicit and implicit contradictions. Moreover, this measure is more precise than the one defined on the basis of attacked arguments, i.e.  $\text{Inc}(a_i, D) = \frac{|\{\beta \in \mathcal{A}_c^{a_i} \text{ such that } \exists (\alpha, \beta) \in \mathcal{R}_c^{a_i}\}|}{|\mathcal{A}_c^{a_i}|}$ . Using this measure, the incoherence degree of the agent in the above example is  $1/3$ . Indeed, even if the argument  $\alpha_1$  is attacked by two arguments, only one conflict is considered.

It is easy to check that if an agent is aggressive towards itself, then it is incoherent.

**Property 3** Let  $D$  be a persuasion dialog, and  $a_i \in \text{Ag}$ . If  $\text{Agr}(a_i, a_i, D) > 0$  then  $\text{Inc}(a_i, D) > 0$ .

The following example shows that the reverse is not always true.

**Example 6** Let  $D$  be a persuasion dialog,  $a_i \in \text{Ag}$ . Let us assume that  $\text{Args}(D^{a_i}) = \{\alpha_1, \alpha_2\}$ , and  $\text{Confs}(D^{a_i}) = \emptyset$ . This means that  $\text{Agr}(a_i, a_i, D) = 0$ . Let  $\text{CAS}_{D^{a_i}} = \langle \mathcal{A}_c^{a_i}, \mathcal{R}_c^{a_i} \rangle$  be the complete version of the previous system with  $\mathcal{A}_c^{a_i} = \{\alpha_1, \alpha_2, \alpha_3\}$  and  $\mathcal{R}_c^{a_i} = \{(\alpha_3, \alpha_1), (\alpha_3, \alpha_2)\}$ . It is clear that  $\text{Inc}(a_i, D) = 2/9$ .

Similarly, it can be shown that if all the formulas of an agent  $a_i$  are borrowed from another agent  $a_j$  and that  $a_i$  is aggressive towards  $a_j$ , then  $a_j$  is for sure incoherent.

**Proposition 4** Let  $D$  be a persuasion dialog, and  $a_i, a_j \in \text{Ag}$ . If  $\text{Loan}(a_i, a_j, D) = 1$  and  $\text{Agr}(a_i, a_j, D) > 0$ , then  $\text{Inc}(a_j, D) > 0$ .

#### 4.3. Measuring the dialog itself

In the previous sections, we have defined measures for evaluating the quality of arguments uttered in a persuasion dialog, and others for analyzing the behavior of agents involved in such a dialog. In this section, we define two other measures for evaluating the quality of the dialog itself. The first measure checks to what extent moves uttered in a given dialog are in relation with the subject of that dialog. It is very common in everyday life, that agents deviate from the subject of the dialog. Before introducing the measure, let us first define formally the notion of relevance.

**Definition 18 (Relevant move)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog. A move  $m_{i=0, \dots, n}$  is relevant to the subject of  $D$  iff there exists a path (not necessarily directed) from  $\text{Subject}(D)$  to  $\text{Content}(m_i)$  in the directed graph associated with  $\text{AS}_D$ .

**Example 4 (continued):** Let us consider the persuasion dialog given in Example 4. Suppose that  $\text{Subject}(D) = \alpha_1$ . It is clear that the arguments  $\alpha_2, \beta_1$  are relevant, whereas the argument  $\beta_2$  is not.

On the basis of this notion of relevance, one can define a measure for knowing the percentage of moves that are relevant in a dialog.

**Definition 19 (Measure of relevance)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog. The relevance degree of  $D$  is

$$\text{Relevance}(D) = \frac{|\{m_{i=0, \dots, n} \text{ such that } m_i \text{ is relevant}\}|}{|D|}$$

**Example 4 (continued):** In the previous example,  $\text{Relevance}(D) = 3/4$ .

It is clear that the greater this degree is, the better the dialog. When the relevance degree of a dialog is equal to 1, this means that agents did not deviate from the subject of the dialog. However, this does not mean at all that all the moves have an impact on the result of the dialog, i.e. on the status of the subject. Another measure is then needed to compute the percentage of useful moves. Before introducing this measure, let us first define what is a useful move. The following definition is similar to the one used in [9].

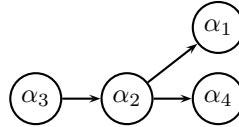
**Definition 20 (Useful move)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog. A move  $m_{i=0, \dots, n}$  is useful in  $D$  iff  $\text{Output}(D) \neq \text{Output}(D \setminus m_i)$  where  $D \setminus m_i$  is the dialog obtained by removing the move  $m_i$ <sup>3</sup>.

It can be checked that the two notions of usefulness and relevance are closely related.

**Proposition 5** Let  $D$  be a persuasion dialog, and  $m \in \mathcal{M}$ . If  $m$  is useful in  $D$ , then  $m$  is relevant to the subject of  $D$ .

Note that the converse is not true as shown in the following example.

**Example 7** Let us assume a dialog  $D$  whose subject is  $\alpha_1$ , and whose graph is the one presented below.



The set  $\{\alpha_1, \alpha_3, \alpha_4\}$  is the only preferred extension of  $\text{AS}_D$ . It is clear that the argument  $\alpha_4$  is relevant to  $\alpha_1$ , but it is not useful for  $D$ . Indeed, the removal of  $\alpha_4$  will not change the status of  $\alpha_1$  which is skeptically accepted.

On the basis of the above notion of usefulness of moves, it is possible to compute the degree of usefulness of the dialog as a whole.

**Definition 21 (Measure of usefulness)** Let  $D = \langle m_0, \dots, m_n \rangle$  be a persuasion dialog. The usefulness degree of  $D$  is

$$\text{Usefulness}(D) = \frac{|\{m_{i=0, \dots, n} \text{ such that } m_i \text{ is useful}\}|}{|D|}$$

It is worth noticing that according to the structure of the graph associated to the argumentation system of a dialog, it is possible to know whether the degrees of relevance and usefulness of that dialog are less than 1 or not. Formally:

**Proposition 6** Let  $D$  be a persuasion dialog, and  $\mathcal{G}$  be the graph associated with  $\text{AS}_D$ . If  $\mathcal{G}$  is not connected then  $\text{Usefulness}(D) < 1$  and  $\text{Relevance}(D) < 1$ .

## 5. Conclusion

Several systems have been proposed in the literature for allowing agents to engage in persuasion dialogs. Different dialog protocols have then been discussed. These latter are the high level rules that govern a dialog. Examples of such rules are "how the turn shifts between agents", and "how moves are chained in a dialog". All these rules should ensure "correct" dialogs, i.e. dialogs that terminate and reach their goals. However, they

---

<sup>3</sup> $D \setminus m_i = \begin{cases} \langle m_0, \dots, m_{i-1}, m_{i+1}, \dots, m_n \rangle & \text{if } i \neq 0 \text{ and } i \neq n \\ \langle m_1, \dots, m_n \rangle & \text{if } i = 0 \\ \langle m_0, \dots, m_{n-1} \rangle & \text{if } i = n \end{cases}$

don't say anything on the *quality* of the dialogs. One even wonders whether there are criteria for measuring the quality of a dialog. In this paper we argue that the answer to this question is yes. Indeed, under the same protocol, different dialogs on the same subject may be generated, and some of them may be judged better than others. There are three kinds of reasons, each of them is translated into quality measures: i) the arguments exchanged are stronger, ii) the generated dialogs are more *concise* (i.e. all the uttered arguments have an impact on the result of the dialog), iii) the behavior of agents was "ideal". In the paper, the behavior of an agent is analyzed on the basis of three main criteria: its *degree of aggressiveness*, its *degree of loan*, and its *degree of coherence*. In sum, different measures have been proposed in this paper for the quality of dialogs. To the best of our knowledge, this is the first work on such measures in dialogs. Exceptions may be the works by Hunter [7] and by Yuan et col. [12] on defining dialog strategies. For instance, Hunter has defined a strategy for selecting arguments in a dialog. The basic idea is that an agent selects the ones that will satisfy the goals of the audience. The agent is thus assumed to maintain two bases: a base containing its own beliefs, and another base containing what the agent thinks are the goals of the audience. These works are thus more concerned with proposing dialog strategies than with analyzing dialogs.

An extension of this work would be the study of the general properties of protocols generating good dialogs w.r.t. the measures presented in this paper. Another future work consists of applying these measures to other types of dialogs, especially negotiation.

## References

- [1] L. Amgoud and C. Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *Int. Journal of Automated Reasoning*, Volume 29 (2):125–169, 2002.
- [2] L. Amgoud and N. Maudet. Strategical considerations for argumentative agents (preliminary report). In *Proceedings of the 10th International Workshop on Non-Monotonic Reasoning NMR'2002 (Collocated with KR'2002), session Argument, Dialogue, Decision*, pages 409–417, 2002.
- [3] L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proceedings of the International Conference on Multi-Agent Systems*, pages 31–38, Boston, MA, 2000.
- [4] L. Black and A. Hunter. A generative inquiry dialogue system. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 963–970. ACM Press, 2007.
- [5] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, 77:321–357, 1995.
- [6] T. F. Gordon. The pleadings game. *Artificial Intelligence and Law*, 2:239–292, 1993.
- [7] A. Hunter. Towards higher impact argumentation. In *Proceedings of the National Conference on Artificial Intelligence*, pages 963–970. ACM Press, 2004.
- [8] S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *Journal of Logic and Computation*, 13(3):347–376, 2003.
- [9] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15:1009–1040, 2005.
- [10] K. Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28(3):203–242, 1990.
- [11] D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, Albany, 1995.
- [12] T. Yuan, D. Moore, and A. Grierson. An assessment of dialogue strategies for a human computer debating system, via computational agents. In *Proc. of ECAI'2004 Workshop on Computational Models of Natural Argument*, pages 17–24, 2004.
- [13] S. Zabala, I. Lara, and H. Geffner. Beliefs, reasons and moves in a model for argumentative dialogues. In *Proc. 25th Latino-American Conf. on Computer Science*, 1999.

# Resolution-Based Argumentation Semantics

Pietro BARONI<sup>a,1</sup> and Massimiliano GIACOMIN<sup>a</sup>

<sup>a</sup> *Dip. Elettronica per l'Automazione, Univ. of Brescia, Italy*

**Abstract.** In a recent work we have proposed a comprehensive set of evaluation criteria for argumentation semantics and we have shown that none of a set of semantics including both traditional and recent proposals is able to meet all criteria. This naturally raises the question whether such criteria are actually satisfiable altogether: this paper provides a positive answer to this question by introducing a new family of argumentation semantics, called *resolution-based* and showing that all the desirable criteria are met by the resolution-based version of grounded semantics.

**Keywords.** Argumentation frameworks, Argumentation semantics, Principle-based evaluation

## Introduction

The issue of defining general evaluation and comparison criteria for argumentation semantics is receiving an increasing attention in recent years [1,2]. In particular, in [2] a comprehensive set of criteria has been introduced and their satisfaction by several literature semantics verified. While some of these criteria correspond to particularly strong properties which are useful for the sake of comparison but are not desirable *per se*, most of them represent generally desirable features for an argumentation semantics. The analysis carried out in [2] shows that none of the “traditional” (stable, complete, grounded, preferred) nor of the more recent (ideal, semi-stable, *CF2*, prudent) semantics is able to satisfy all the desirable criteria. An interesting question, then, is whether such criteria can be satisfied altogether or their satisfaction is prevented by some inherent impossibility constraint. We provide an answer to this question by examining a scheme of semantics definition called “resolution-based”: given a “conventional” semantics  $S$ , its resolution-based version  $S^*$  is defined by resorting to the notion of *resolutions* of an argumentation framework [3,2]. We examine the resolution-based versions of two traditional (grounded and preferred) semantics and show in particular that the resolution-based version of grounded semantics is able to satisfy all the desirable criteria considered in [2].

---

<sup>1</sup>Corresponding Author: Pietro Baroni, Dip. Elettronica per l'Automazione, Univ. of Brescia, Via Branze 38, 25123 Brescia, Italy. Tel.: +39 030 3715455; Fax: +39 030 380014; E-mail: baroni@ing.unibs.it.

## 1. Background Concepts and Notation

The present work lies in the frame of the general theory of abstract argumentation frameworks proposed by Dung [4].

**Definition 1** An argumentation framework is a pair  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , where  $\mathcal{A}$  is a set, and  $\rightarrow \subseteq (\mathcal{A} \times \mathcal{A})$  is a binary relation on  $\mathcal{A}$ , called attack relation.

In the following we will always assume that  $\mathcal{A}$  is finite. The arguments attacking a given argument  $\alpha$  are called *defeaters* (or *parents*) of  $\alpha$  and form a set which is denoted as  $\text{par}_{\text{AF}}(\alpha)$ .

**Definition 2** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  and an argument  $\alpha \in \mathcal{A}$ ,  $\text{par}_{\text{AF}}(\alpha) \triangleq \{\beta \in \mathcal{A} \mid \beta \rightarrow \alpha\}$ . If  $\text{par}_{\text{AF}}(\alpha) = \emptyset$ , then  $\alpha$  is called an initial argument. The set of initial arguments of  $\text{AF}$  is denoted as  $\text{IN}(\text{AF})$ .

Since we will frequently deal with sets of arguments, it is useful to define suitable notations for them.

**Definition 3** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , an argument  $\alpha \in \mathcal{A}$  and two (not necessarily disjoint) sets  $S, P \subseteq \mathcal{A}$ , we define:  $S \rightarrow \alpha \equiv \exists \beta \in S : \beta \rightarrow \alpha$ ;  $\alpha \rightarrow S \equiv \exists \beta \in S : \alpha \rightarrow \beta$ ;  $S \rightarrow P \equiv \exists \alpha \in S, \beta \in P : \alpha \rightarrow \beta$ .

We also define the *restriction* of an argumentation framework to a subset  $S$  of its arguments.

**Definition 4** Let  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  be an argumentation framework. The restriction of  $\text{AF}$  to  $S \subseteq \mathcal{A}$  is the argumentation framework  $\text{AF} \downarrow_S = \langle S, \rightarrow \cap (S \times S) \rangle$ .

In Dung's theory, an argumentation semantics is defined by specifying the criteria for deriving, given a generic argumentation framework, the set of all possible extensions, each one representing a set of arguments considered to be acceptable together. Accordingly, a basic requirement for any extension  $E$  is that it is *conflict-free*, namely  $\nexists \alpha, \beta \in E : \alpha \rightarrow \beta$ , denoted in the following as  $cf(E)$ . All argumentation semantics proposed in the literature satisfy this fundamental *conflict-free property*. The set of maximal (with respect to set inclusion) conflict free sets of  $\text{AF}$  will be denoted as  $\mathcal{MCF}_{\text{AF}}$ .

Given a generic argumentation semantics  $\mathcal{S}$ , the set of extensions prescribed by  $\mathcal{S}$  for a given argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  will be denoted as  $\mathcal{E}_{\mathcal{S}}(\text{AF})$ . If it holds that  $\forall \text{AF} \mid \mathcal{E}_{\mathcal{S}}(\text{AF}) \mid = 1$ , then the semantics  $\mathcal{S}$  is said to follow the *unique-status approach*, otherwise it is said to follow the *multiple-status approach*. A relevant question concerns the existence of extensions. For a given semantics  $\mathcal{S}$ , we define  $\mathcal{D}_{\mathcal{S}} = \{\text{AF} \mid \mathcal{E}_{\mathcal{S}}(\text{AF}) \neq \emptyset\}$ , namely the set of argumentation frameworks where  $\mathcal{S}$  admits at least one extension. If  $\forall \text{AF} \mid \text{AF} \in \mathcal{D}_{\mathcal{S}} \mid$  we will say that  $\mathcal{S}$  is *universally defined*. Most literature semantics are universally defined, with the notable exception of stable semantics [4].

In the following we assume that the reader is familiar with the traditional grounded, complete, and preferred semantics, denoted as  $\mathcal{GR}$ ,  $\mathcal{CO}$ ,  $\mathcal{PR}$  respectively. Their definition [4] can not be recalled here due to space limitation. It is well-known that grounded semantics belongs to the unique-status approach: the grounded extension of an argumentation framework  $\text{AF}$  will be denoted as  $\text{GE}(\text{AF})$ .

## 2. A Review of Semantics Evaluation Criteria

We quickly recall the definition of semantics evaluation criteria discussed in [2] to which the reader is referred for more details.

### 2.1. Extension Evaluation Criteria

The I-maximality criterion states that no extension is a proper subset of another one.

**Definition 5** A set of extensions  $\mathcal{E}$  is I-maximal iff  $\forall E_1, E_2 \in \mathcal{E}$ , if  $E_1 \subseteq E_2$  then  $E_1 = E_2$ . A semantics  $\mathcal{S}$  satisfies the I-maximality criterion if and only if  $\forall \text{AF}, \mathcal{E}_{\mathcal{S}}(\text{AF})$  is I-maximal.

The requirement of *admissibility* lies at the heart of all semantics discussed in [4] and is based on the notions of acceptable argument and admissible set, the underlying idea being that an extension should be able to defend itself against attacks.

**Definition 6** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , an argument  $\alpha \in \mathcal{A}$  is acceptable with respect to a set  $E \subseteq \mathcal{A}$  if and only if  $\forall \beta \in \mathcal{A} : \beta \rightarrow \alpha, E \rightarrow \beta$ . A set  $E \subseteq \mathcal{A}$  is admissible if and only if  $E$  is conflict-free and any argument  $\alpha \in E$  is acceptable with respect to  $E$ . The set of the admissible sets of AF will be denoted as  $\mathcal{AS}(\text{AF})$ .

**Definition 7** A semantics  $\mathcal{S}$  satisfies the admissibility criterion if  $\forall \text{AF} \in \mathcal{D}_{\mathcal{S}}, \forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF}) E \in \mathcal{AS}(\text{AF})$ , namely:

$$\alpha \in E \Rightarrow \forall \beta \in \text{par}_{\text{AF}}(\alpha), E \rightarrow \beta \quad (1)$$

Condition (1) includes the case where  $\alpha$  defends itself against (some of) its defeaters. A stronger notion of defense may also be considered where an argument  $\alpha$  cannot defend itself nor can be involved in its own defense. To formalize this requirement the notion of *strongly defended argument* has been introduced in [2].

**Definition 8** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ ,  $\alpha \in \mathcal{A}$  and  $S \subseteq \mathcal{A}$ ,  $\alpha$  is strongly defended by  $S$  (denoted as  $sd(\alpha, S)$ ) iff  $\forall \beta \in \text{par}_{\text{AF}}(\alpha), \exists \gamma \in S \setminus \{\alpha\} : \gamma \rightarrow \beta$  and  $sd(\gamma, S \setminus \{\alpha\})$ .

In words,  $\alpha$  is strongly defended by  $S$  if  $S$  includes a defeater  $\gamma \neq \alpha$  for any defeater  $\beta$  of  $\alpha$ . In turn,  $\gamma$  has to be strongly defended by  $S \setminus \alpha$ , namely  $\gamma$  needs neither  $\alpha$  nor itself to be defended against its defeaters in AF. The recursion is well founded since, at any step, a set of strictly lesser cardinality is considered. In particular, if  $sd(\alpha, S)$  then the base of this recursive definition is provided by initial arguments, which are strongly defended by any set since they have no defeaters. The notion of strong defense is the basis for the definition of the *strong admissibility criterion*.

**Definition 9** A semantics  $\mathcal{S}$  satisfies the strong admissibility criterion, if  $\forall \text{AF} \in \mathcal{D}_{\mathcal{S}}, \forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF})$  it holds that:

$$\alpha \in E \Rightarrow sd(\alpha, E) \quad (2)$$

Strong admissibility can be regarded as a specific feature of grounded semantics (no other semantics considered in [2] satisfies this property except the *prudent* version of grounded semantics). Indeed it is shown in [2] that the conjunction of strong admissibility and weak reinstatement (defined below and satisfied by all semantics considered in [2] except several *prudent* semantics) provides an equivalent characterization of grounded semantics, therefore only grounded semantics is able to satisfy both these criteria. For these reasons, strong admissibility should not be included in the set of generally desirable properties for an argumentation semantics.

The property of *reinstatement* corresponds to the converse of the implication (1) prescribed by the admissibility criterion. Intuitively, an argument  $\alpha$  is *reinstated* if its defeaters are in turn defeated and, as a consequence, one may assume that they should have no effect on the extension membership of  $\alpha$  (and hence on its justification state). Under this assumption, if an extension  $E$  reinstates  $\alpha$  then  $\alpha$  should belong to  $E$ . Formally this leads to the following *reinstatement criterion*:

**Definition 10** A semantics  $\mathcal{S}$  satisfies the *reinstatement criterion* if  $\forall \text{AF} \in \mathcal{D}_{\mathcal{S}}, \forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF})$  it holds that:

$$(\forall \beta \in \text{par}_{\text{AF}}(\alpha), E \rightarrow \beta) \Rightarrow \alpha \in E \quad (3)$$

Considering the strong notion of defense we obtain a *weak* (since it is implied by condition (3)) *reinstatement criterion*.

**Definition 11** A semantics  $\mathcal{S}$  satisfies the *weak reinstatement criterion* if  $\forall \text{AF} \in \mathcal{D}_{\mathcal{S}}, \forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF})$  it holds that:

$$sd(\alpha, E) \Rightarrow \alpha \in E \quad (4)$$

Another observation concerns the fact that condition (3) prescribes that an argument  $\alpha$  defended by an extension should be included in the extension, without specifying that  $\alpha$  should not give rise to conflicts within the extension. To explicitly take into account this aspect, the following *CF-reinstatement criterion* can be given.

**Definition 12** A semantics  $\mathcal{S}$  satisfies the *CF-reinstatement criterion* if  $\forall \text{AF} \in \mathcal{D}_{\mathcal{S}}, \forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF})$  it holds that:

$$((\forall \beta \in \text{par}_{\text{AF}}(\alpha) E \rightarrow \beta) \wedge cf(E \cup \{\alpha\})) \Rightarrow \alpha \in E \quad (5)$$

The notion of *directionality* is based on the idea that the extension membership of an argument  $\alpha$  should be affected only by the defeaters of  $\alpha$  (which in turn are affected by their defeaters and so on), while the arguments which only receive an attack from  $\alpha$  (and in turn those which are attacked by them and so on) should not have any effect on  $\alpha$ . The directionality criterion can be specified by requiring that an unattacked set of arguments is not affected by the remaining parts of the argumentation framework as far as extensions are concerned.

**Definition 13** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , a set  $U \subseteq \mathcal{A}$  is unattacked if and only if  $\exists \alpha \in (\mathcal{A} \setminus U) : \alpha \rightarrow U$ . The set of unattacked sets of  $\text{AF}$  will be denoted as  $US(\text{AF})$ .

**Definition 14** A semantics  $\mathcal{S}$  satisfies the directionality criterion if and only if  $\forall \text{AF} = \langle \mathcal{A}, \rightarrow \rangle, \forall U \in \mathcal{US}(\text{AF}), \mathcal{AE}_{\mathcal{S}}(\text{AF}, U) = \mathcal{E}_{\mathcal{S}}(\text{AF} \downarrow_U)$  where  $\mathcal{AE}_{\mathcal{S}}(\text{AF}, S) = \{(E \cap S) \mid E \in \mathcal{E}_{\mathcal{S}}(\text{AF})\} \subseteq 2^{\mathcal{A}}$ .

In words, the intersection of any extension prescribed by  $\mathcal{S}$  for  $\text{AF}$  with an unattacked set  $U$  is equal to one of the extensions prescribed by  $\mathcal{S}$  for the restriction of  $\text{AF}$  to  $U$ , and vice versa.

## 2.2. Skepticism Related Criteria

Semantics *adequacy* criteria introduced in [2] are based on comparisons of sets of extensions which in turn exploit some recently introduced notions concerning the formalization of *skepticism*.

### 2.2.1. Skepticism Relations

The notion of skepticism has often been used in informal or semi-formal ways to discuss semantics behavior, e.g. by observing that a semantics  $\mathcal{S}_1$  is “more skeptical” than another semantics  $\mathcal{S}_2$ , which intuitively means that  $\mathcal{S}_1$  makes less committed choices than  $\mathcal{S}_2$  about the justification state of the arguments. While the issue of skepticism has been considered in the literature mostly in the case of comparison between specific proposals, a more general analysis has been carried out in [2] and is partly recalled here. First, we consider as a basic concept a generic relation of skepticism  $\preceq^E$  between sets of extensions: given two sets of extensions  $\mathcal{E}_1, \mathcal{E}_2$  of an argumentation framework  $\text{AF}$ ,  $\mathcal{E}_1 \preceq^E \mathcal{E}_2$  will simply denote that  $\mathcal{E}_1$  is at least as skeptical as  $\mathcal{E}_2$  in some sense. In the approach to semantics evaluation we are recalling, a skepticism relation is used to compare the sets of extensions prescribed by a particular semantics on different but related argumentation frameworks. To this purpose, one first needs to define a skepticism relation between argumentation frameworks based on the same set of arguments: given two argumentation frameworks  $\text{AF}_1 = \langle \mathcal{A}, \rightarrow_1 \rangle$  and  $\text{AF}_2 = \langle \mathcal{A}, \rightarrow_2 \rangle$ ,  $\text{AF}_1 \preceq^A \text{AF}_2$  denotes that  $\text{AF}_1$  (actually its attack relation) is inherently less committed (to be precise, not more committed) than  $\text{AF}_2$ . Then, one may reasonably require that any semantics reflects in its extensions the skepticism relations between argumentation frameworks. Requirements of this kind for a generic semantics  $\mathcal{S}$  will be called *adequacy criteria*. Having laid out the general framework for the definition of adequacy criteria, we first recall the actual skepticism relations we will use in the sequel.

Let us start, at a basic level, by noting that defining a relation of skepticism between two extensions is intuitively straightforward: an extension  $E_1$  is more skeptical than an extension  $E_2$  if and only if  $E_1 \subseteq E_2$ . In fact, a more skeptical attitude corresponds to a smaller set of selected arguments. Directly extending the above intuition to the comparison of sets of extensions leads to define the following skepticism relation  $\preceq_{\cap}^E$ .

**Definition 15** Given two sets of extensions  $\mathcal{E}_1$  and  $\mathcal{E}_2$  of an argumentation framework  $\text{AF}$ ,  $\mathcal{E}_1 \preceq_{\cap}^E \mathcal{E}_2$  iff  $\bigcap_{E_1 \in \mathcal{E}_1} E_1 \subseteq \bigcap_{E_2 \in \mathcal{E}_2} E_2$

Finer (and actually stronger) skepticism relations can then be defined by considering relations of pairwise inclusion between extensions. We recall that to compare a single extension  $E_1$  with a set of extensions  $\mathcal{E}_2$ , the relation  $\forall E_2 \in \mathcal{E}_2, E_1 \subseteq E_2$  has often been used in the literature (for instance to verify that the unique extension prescribed

by grounded semantics is more skeptical than the set of extensions prescribed by preferred semantics). A direct generalization to the comparison of two sets of extensions is represented by the following weak skepticism relation  $\preceq_W^E$ .

**Definition 16** Given two sets of extensions  $\mathcal{E}_1$  and  $\mathcal{E}_2$  of an argumentation framework AF,  $\mathcal{E}_1 \preceq_W^E \mathcal{E}_2$  iff

$$\forall E_2 \in \mathcal{E}_2 \exists E_1 \in \mathcal{E}_1 : E_1 \subseteq E_2 \quad (6)$$

It is worth noting that (as it is easy to see), given two sets of extensions  $\mathcal{E}_1$  and  $\mathcal{E}_2$  of an argumentation framework AF,  $\mathcal{E}_1 \preceq_W^E \mathcal{E}_2 \Rightarrow \mathcal{E}_1 \preceq_{\cap}^E \mathcal{E}_2$ .

In a sense, relation  $\preceq_W^E$  is unidirectional, since it only constrains the extensions of  $\mathcal{E}_2$ , while  $\mathcal{E}_1$  may contain additional extensions unrelated to those of  $\mathcal{E}_2$ . One may then consider also a more symmetric (and stronger) relationship  $\preceq_S^E$ , where it is also required that any extension of  $\mathcal{E}_1$  is included in one extension of  $\mathcal{E}_2$ . However, as discussed in [5,6,2] this relationship is definitely too strong since it actually prevents comparability of any pair of multiple-status semantics. For this reason, it will not be considered here.

Turning to skepticism relations between argumentation frameworks, a relation  $\preceq^A$  has been proposed in [2], generalizing some more specific but related notions introduced in [6] and [3]. It relies on a basic intuition concerning the relationship between a mutual and a unidirectional attack between two arguments  $\alpha$  and  $\beta$ . To illustrate this intuition, let us consider a very simple argumentation framework including just two arguments  $\alpha$  and  $\beta$ , where  $\alpha$  attacks  $\beta$  but not vice versa. This is a situation where the status assignment of any argumentation semantics corresponds to the maximum level of commitment: it is universally accepted that  $\alpha$  should be justified and  $\beta$  rejected. Now if we consider a modified argumentation framework where an attack from  $\beta$  to  $\alpha$  has been added, we obtain a situation where, clearly, a lesser level of commitment is appropriate: given the mutual attack between the two arguments, neither of them can be assigned a definitely committed status and we are left in a more undecided, i.e. more skeptical, situation in absence of any reason for preferring either of them. Extending this reasoning, consider a couple of arguments  $\alpha$  and  $\beta$  in a generic argumentation framework AF such that  $\alpha \rightarrow \beta$  while  $\beta \not\rightarrow \alpha$ . Consider now an argumentation framework  $AF'$  obtained from AF by simply adding an attack relation from  $\beta$  to  $\alpha$  while leaving all the rest unchanged. It seems reasonable to state that  $AF'$  corresponds to a more undecided situation with respect to AF. This reasoning can be generalized by considering any number of “transformations” of unidirectional attacks into mutual ones.

Dealing with the case of hierarchical argumentation frameworks but relying on the same intuition, Modgil has independently introduced in [3] the notion of *partial resolution* of an argumentation framework AF, which, roughly speaking, is an argumentation framework  $AF'$  where some mutual attacks of AF are converted into unidirectional ones.  $AF'$  is a (complete) *resolution* of AF if all mutual attacks of AF are converted. Also in [3], the underlying idea is that the presence of mutual attacks corresponds to a more undecided situation. While more general notions of resolution (e.g. involving cycles of any length, rather than mutual attacks only) might be considered, this is not necessary to obtain the main results of the paper and is left for future work.

We are now ready to define a skepticism relation  $\preceq^A$  between argumentation frameworks based on the same set of arguments, which requires a preliminary notation concerning the set of conflicting pairs of arguments in an argumentation framework.

**Definition 17** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ ,  $\mathcal{CONF}(\text{AF}) \triangleq \{(\alpha, \beta) \in \mathcal{A} \times \mathcal{A} \mid \alpha \rightarrow \beta \vee \beta \rightarrow \alpha\}$

**Definition 18** Given two argumentation frameworks  $\text{AF}_1 = \langle \mathcal{A}, \rightarrow_1 \rangle$  and  $\text{AF}_2 = \langle \mathcal{A}, \rightarrow_2 \rangle$ ,  $\text{AF}_1 \preceq^A \text{AF}_2$  if and only if  $\mathcal{CONF}(\text{AF}_1) = \mathcal{CONF}(\text{AF}_2)$  and  $\rightarrow_2 \subseteq \rightarrow_1$ .

It is easy to see that the above definition covers all cases where some (possibly none) mutual attacks of  $\text{AF}_1$  correspond to unidirectional attacks in  $\text{AF}_2$ , while unidirectional attacks of  $\text{AF}_1$  are the same in  $\text{AF}_2$  (using the terminology of [3],  $\text{AF}_2$  is a partial resolution of  $\text{AF}_1$ ). It is also immediate to see that  $\preceq^A$  is a partial order, as it consists of an equality and a set-inclusion relation. Comparable argumentation frameworks are characterized by having the same set of arguments and the same set of conflicting pairs of arguments. In this respect, it is also worth noting that within a set of comparable argumentation frameworks there are, in general, several maximal elements with respect to  $\preceq^A$ , namely all argumentation frameworks where no mutual attack is present (corresponding to resolutions, in the terminology of [3]). Given an argumentation framework  $\text{AF}$ , in the following we will denote as  $\mathcal{RES}(\text{AF})$  the set of argumentation frameworks comparable with  $\text{AF}$  and maximal with respect to  $\preceq^A$ .

### 2.2.2. Skepticism Adequacy

Given that an argumentation framework is considered inherently more skeptical than another one, it is reasonable to require that when applying the same semantics to both, the skepticism relation between them is preserved between their sets of extensions. This kind of criterion, called *skepticism adequacy*, has first been proposed in [6] and is formulated here in a generalized version.

**Definition 19** Given a skepticism relation  $\preceq^E$  between sets of extensions, a semantics  $\mathcal{S}$  is  $\preceq^E$ -skepticism-adequate, denoted  $\mathcal{SA}_{\preceq^E}(\mathcal{S})$ , if and only if for any pair of argumentation frameworks  $\text{AF}, \text{AF}'$  such that  $\text{AF} \preceq^A \text{AF}'$  it holds that  $\mathcal{E}_{\mathcal{S}}(\text{AF}) \preceq^E \mathcal{E}_{\mathcal{S}}(\text{AF}')$ .

According to the definitions provided in Section 2.2.1 we have two skepticism adequacy properties, which are clearly related by the same order of implication:  $\mathcal{SA}_{\preceq^E_W}(\mathcal{S}) \Rightarrow \mathcal{SA}_{\preceq^E_{\cap}}(\mathcal{S})$ .

### 2.2.3. Resolution Adequacy

Resolution adequacy generalizes a criterion first proposed in [3] and relies on the intuition that if an argument is included in all extensions of all possible resolutions of an argumentation framework  $\text{AF}$  then it should be included in all extensions of  $\text{AF}$  too. This criterion is called *resolution adequacy* in [2] where a generalization of its original formulation is provided, in order to make it parametric with respect to skepticism relations between sets of extensions.

**Definition 20** Given a skepticism relation  $\preceq^E$  between sets of extensions, a semantics  $\mathcal{S}$  is  $\preceq^E$ -resolution-adequate, denoted  $\mathcal{RA}_{\preceq^E}(\mathcal{S})$ , if and only if for any argumentation framework  $\text{AF}$  it holds that  $\mathcal{UR}(\text{AF}, \mathcal{S}) \preceq^E \mathcal{E}_{\mathcal{S}}(\text{AF})$ , where  $\mathcal{UR}(\text{AF}, \mathcal{S}) = \bigcup_{\text{AF}' \in \mathcal{RES}(\text{AF})} \mathcal{E}_{\mathcal{S}}(\text{AF}')$ .

Again, we have two resolution adequacy properties, related by the usual order of implication:  $\mathcal{RA}_{\preceq^E_W}(\mathcal{S}) \Rightarrow \mathcal{RA}_{\preceq^E_{\cap}}(\mathcal{S})$ .

### 3. Defining and Evaluating Resolution-Based Semantics

Pursuing the goal of verifying whether the desirable properties discussed in Section 2 (namely I-maximality, admissibility, reinstatement, weak reinstatement,  $\mathcal{CF}$ -reinstatement, directionality,  $\preceq_{\cap}^E$ -and  $\preceq_W^E$ -skepticism adequacy,  $\preceq_{\cap}^E$ - and  $\preceq_W^E$ -resolution adequacy) can be satisfied altogether, we introduce a parametric family of semantics, called *resolution-based* since its definition is based on the notion of resolutions of an argumentation framework.

**Definition 21** *Given an argumentation semantics  $\mathcal{S}$  which is universally defined, its resolution-based version is the semantics  $\mathcal{S}^*$  such that for any argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$   $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) = \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{S}))$ , where given a set  $\mathcal{E}$  of subsets of  $\mathcal{A}$ ,  $\mathcal{MIN}(\mathcal{E})$  denotes the set of the minimal (with respect to set inclusion) elements of  $\mathcal{E}$ .*

Operationally, the idea underlying Definition 21 is as follows: given an argumentation framework  $\text{AF}$  the set  $\mathcal{RES}(\text{AF})$  of its resolutions (i.e. of all argumentation frameworks derivable from  $\text{AF}$  by transforming mutual attacks into unidirectional ones) is determined. Then semantics  $\mathcal{S}$  is applied to each  $\text{AF}' \in \mathcal{RES}(\text{AF})$  to obtain  $\mathcal{E}_{\mathcal{S}}(\text{AF}')$ . The union  $\mathcal{UR}(\text{AF}, \mathcal{S})$  of these sets of extensions is then considered and its minimal elements (with respect to set inclusion) selected as the extensions prescribed for  $\text{AF}$  by  $\mathcal{S}^*$ . Note that this definition directly enforces the property of I-maximality.

Let us now show that for any semantics  $\mathcal{S}$ ,  $\mathcal{S}^*$  satisfies  $\preceq_W^E$ -skepticism adequacy (and therefore  $\preceq_{\cap}^E$ -skepticism adequacy).

**Proposition 1** *For every argumentation semantics  $\mathcal{S}$ , its resolution based version  $\mathcal{S}^*$  satisfies  $\preceq_W^E$ -skepticism adequacy.*

*Proof.* On the basis of Definition 19 we have to show that for any pair of argumentation frameworks  $\text{AF}, \text{AF}'$  such that  $\text{AF} \preceq^A \text{AF}'$  it holds that  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \preceq_W^E \mathcal{E}_{\mathcal{S}^*}(\text{AF}')$ . First, it is easy to see that for any such a pair of argumentation frameworks  $\mathcal{RES}(\text{AF}') \subseteq \mathcal{RES}(\text{AF})$ , and, therefore, by definition of  $\mathcal{UR}$ ,  $\mathcal{UR}(\text{AF}', \mathcal{S}) \subseteq \mathcal{UR}(\text{AF}, \mathcal{S})$ . Recalling that  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) = \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{S}))$  and  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}') = \mathcal{MIN}(\mathcal{UR}(\text{AF}', \mathcal{S}))$ , it follows that  $\forall E' \in \mathcal{E}_{\mathcal{S}^*}(\text{AF}') \exists E \in \mathcal{UR}(\text{AF}, \mathcal{S})$  and hence  $\exists E \in \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{S})) = \mathcal{E}_{\mathcal{S}^*}(\text{AF})$  such that  $E \subseteq E'$ , namely  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \preceq_W^E \mathcal{E}_{\mathcal{S}^*}(\text{AF}')$ .  $\square$

Adding the (not particularly restrictive) hypothesis that  $\mathcal{S}$  is I-maximal,  $\mathcal{S}^*$  achieves also the property of  $\preceq_W^E$ - (and  $\preceq_{\cap}^E$ ) resolution adequacy, as shown by the following proposition.

**Proposition 2** *For every I-maximal argumentation semantics  $\mathcal{S}$ , its resolution based version  $\mathcal{S}^*$  satisfies  $\preceq_W^E$ -resolution adequacy.*

*Proof.* According to Definition 20 we have to show that  $\mathcal{UR}(\text{AF}, \mathcal{S}^*) \preceq_W^E \mathcal{E}_{\mathcal{S}^*}(\text{AF})$ . By definition of  $\mathcal{UR}$  we have  $\mathcal{UR}(\text{AF}, \mathcal{S}^*) = \bigcup_{\text{AF}' \in \mathcal{RES}(\text{AF})} \mathcal{E}_{\mathcal{S}^*}(\text{AF}')$ . In turn, by Definition 21,  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}') = \mathcal{MIN}(\mathcal{UR}(\text{AF}', \mathcal{S})) = \mathcal{MIN}(\bigcup_{\text{AF}'' \in \mathcal{RES}(\text{AF}')} \mathcal{E}_{\mathcal{S}}(\text{AF}''))$ . Since any  $\text{AF}'$  belongs to  $\mathcal{RES}(\text{AF})$ , it is immediate to see that  $\mathcal{RES}(\text{AF}') = \{\text{AF}'\}$  therefore  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}') = \mathcal{MIN}(\mathcal{E}_{\mathcal{S}}(\text{AF}')) = \mathcal{E}_{\mathcal{S}}(\text{AF}')$ , where the last equality holds by the hypothesis of I-maximality of  $\mathcal{S}$ . It follows that  $\mathcal{UR}(\text{AF}, \mathcal{S}^*) = \bigcup_{\text{AF}' \in \mathcal{RES}(\text{AF})} \mathcal{E}_{\mathcal{S}}(\text{AF}') = \mathcal{UR}(\text{AF}, \mathcal{S})$ . On the other hand, by definition  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) =$

$\mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{S}))$ . It follows  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \subseteq \mathcal{UR}(\text{AF}, \mathcal{S}) = \mathcal{UR}(\text{AF}, \mathcal{S}^*)$  which directly implies  $\mathcal{UR}(\text{AF}, \mathcal{S}^*) \preceq_{\mathcal{W}}^{\mathcal{S}} \mathcal{E}_{\mathcal{S}^*}(\text{AF})$ .  $\square$

Turning to defense related criteria, a significant result can be obtained using as a basis the following lemma concerning the relationship between the complete extensions of the resolutions of AF and those of AF itself.

**Lemma 1** *Considering an argumentation framework AF, if  $\exists \text{AF}' \in \mathcal{RES}(\text{AF})$  such that a set E is a complete extension in  $\text{AF}'$  then E is also a complete extension in AF.*

*Proof.* It is shown in Lemma 1 of [3] that for any argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle \mathcal{AS}(\text{AF}) = \bigcup_{\text{AF}' \in \mathcal{RES}(\text{AF})} \mathcal{AS}(\text{AF}')$ . This entails that E is admissible in AF, therefore we have only to prove that it is a complete extension, i.e. that  $\forall \alpha$ , if  $\forall \beta \in \text{par}_{\text{AF}}(\alpha) E \rightarrow \beta$  then  $\alpha \in E$ . Since according to Definition 18  $\mathcal{CONF}(\text{AF}') = \mathcal{CONF}(\text{AF})$ , it must be the case that  $\forall \beta : E \rightarrow \beta$  in AF,  $\exists \gamma \in E$  such that in  $\text{AF}'$  either  $\gamma \rightarrow \beta$  or  $\beta \rightarrow \gamma$ . On the other hand, since E is admissible in  $\text{AF}'$  if  $\beta \rightarrow \gamma$  in  $\text{AF}'$  then also  $E \rightarrow \beta$  holds in  $\text{AF}'$ . In sum, we have that  $\forall \beta \in \text{par}_{\text{AF}}(\alpha)$ ,  $E \rightarrow \beta$  holds in  $\text{AF}'$ . Now, since no additional attack edges are present in  $\text{AF}'$  with respect to those of AF it must be the case that  $\text{par}_{\text{AF}'}(\alpha) \subseteq \text{par}_{\text{AF}}(\alpha)$ . As a consequence,  $\forall \gamma \in \text{par}_{\text{AF}'}(\alpha)$   $E \rightarrow \gamma$  holds in  $\text{AF}'$ , and since E is a complete extension in  $\text{AF}'$  then  $\alpha \in E$ .  $\square$

**Proposition 3** *If a semantics  $\mathcal{S}$  is such that for any argumentation framework AF  $\mathcal{E}_{\mathcal{S}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF})$ , then also  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF})$ .*

*Proof.*  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \subseteq \mathcal{UR}(\text{AF}, \mathcal{S}) = \bigcup_{\text{AF}' \in \mathcal{RES}(\text{AF})} \mathcal{E}_{\mathcal{S}}(\text{AF}')$ . By the hypothesis, for any  $\text{AF}' \in \mathcal{RES}(\text{AF})$   $\mathcal{E}_{\mathcal{S}}(\text{AF}') \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF}')$  and, by Lemma 1,  $\mathcal{E}_{\mathcal{CO}}(\text{AF}') \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF})$ . It directly follows that  $\mathcal{E}_{\mathcal{S}^*}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF})$ .  $\square$

Since it is known [2] that complete extensions satisfy the property of admissibility (1) and reinstatement (3) (which in turn entails weak (4) and  $\mathcal{CF}$  (5) reinstatement), then any  $\mathcal{S}^*$  satisfying the hypothesis of Proposition 3 also satisfies these properties. Since both grounded and preferred semantics are I-maximal (see [2]) and satisfy the hypothesis of Proposition 3, the results presented above show that the resolution-based versions of grounded and preferred semantics satisfy all the desirable requirements listed at the beginning of this section, except directionality.

Proposition 4 provides the final result by showing that directionality is satisfied by the resolution-based version of grounded semantics, denoted as  $\mathcal{GR}^*$ . A preliminary lemma is needed.

**Lemma 2** *For any argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  and for any set  $I \subseteq \mathcal{A}$ ,  $\{\text{AF}' \downarrow_I \mid \text{AF}' \in \mathcal{RES}(\text{AF})\} = \mathcal{RES}(\text{AF} \downarrow_I)$ .*

*Proof.* Let us first prove that, given  $\text{AF}' \in \mathcal{RES}(\text{AF})$ ,  $\text{AF}' \downarrow_I \in \mathcal{RES}(\text{AF} \downarrow_I)$ . Obviously,  $\text{AF}' \downarrow_I$  and  $\text{AF} \downarrow_I$  have the same set of arguments, namely  $I$ . Moreover, for any  $\alpha, \beta \in I$  the edges between  $\alpha$  and  $\beta$  of  $\text{AF}' \downarrow_I$  are those of  $\text{AF}'$ , and the same holds for the edges of  $\text{AF} \downarrow_I$  with respect of those of AF. Since  $\text{AF}' \in \mathcal{RES}(\text{AF})$ , this entails that  $\mathcal{CONF}(\text{AF}' \downarrow_I) = \mathcal{CONF}(\text{AF} \downarrow_I)$  and that all the edges of  $\text{AF}' \downarrow_I$  also belong to  $\text{AF} \downarrow_I$ , therefore  $\text{AF} \downarrow_I \preceq^A \text{AF}' \downarrow_I$ . Finally, since  $\text{AF}'$  does not contain mutual attacks the same holds for  $\text{AF}' \downarrow_I$ , i.e.  $\text{AF}' \downarrow_I$  is maximal with respect to  $\preceq^A$ .

Turning to the other direction of the proof, given a generic  $\text{AF}'' \in \mathcal{RES}(\text{AF} \downarrow_I)$  we

have to prove that there is  $\text{AF}' \in \mathcal{RES}(\text{AF})$  such that  $\text{AF}' \downarrow_I = \text{AF}''$ .  $\text{AF}'$  can be constructed from  $\text{AF}$  by selecting a unidirectional attack for each mutual attack of  $\text{AF}$ , with the constraint that  $\forall \alpha, \beta \in I$  the chosen edge between  $\alpha$  and  $\beta$  is that of  $\text{AF}''$  (this is possible since  $\text{AF}'' \in \mathcal{RES}(\text{AF} \downarrow_I)$ ). It is then easy to see that  $\text{AF}' \in \mathcal{RES}(\text{AF})$  and that  $\text{AF}' \downarrow_I = \text{AF}''$ .  $\square$

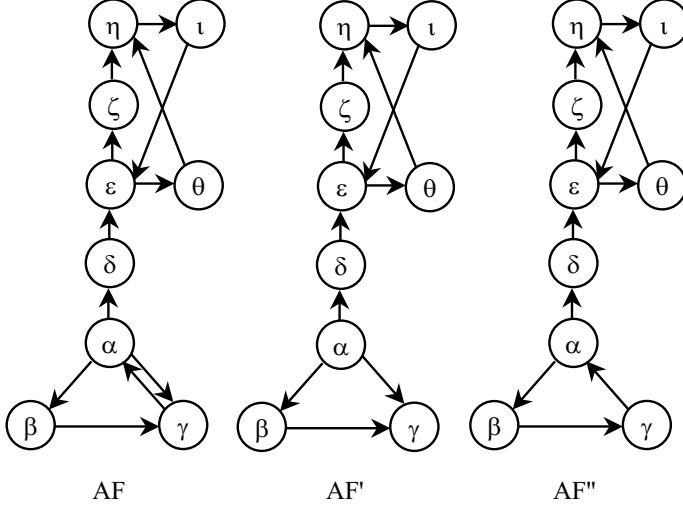
**Proposition 4**  $\mathcal{GR}^*$  satisfies the directionality property.

*Proof.* According to Definition 14, we have to prove that  $\forall \text{AF}, \forall U \in \mathcal{US}(\text{AF}) \{ (E \cap U) \mid E \in \mathcal{EGR}^*(\text{AF}) \} = \mathcal{EGR}^*(\text{AF} \downarrow_U)$ .

Let  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  as usual. First, we prove that given a set  $E \in \mathcal{EGR}^*(\text{AF})$  ( $E \cap U \in \mathcal{EGR}^*(\text{AF} \downarrow_U)$ ). Since  $E \in \mathcal{EGR}^*(\text{AF})$ , according to Definition 21  $\exists \text{AF}' \in \mathcal{RES}(\text{AF})$  such that  $E \in \mathcal{EGR}(\text{AF}')$ . Since grounded semantics belongs to the unique status approach this is equivalent to  $\exists \text{AF}' \in \mathcal{RES}(\text{AF})$  such that  $E = \text{GE}(\text{AF}')$ . It is easy to see that the set  $U$ , being unattacked in  $\text{AF}$ , also belongs to  $\mathcal{US}(\text{AF}')$ . Since grounded semantics satisfies the directionality property [2], we can derive  $(E \cap U) = \text{GE}(\text{AF}' \downarrow_U)$ . By Lemma 2, we have  $\text{AF}' \downarrow_U \in \mathcal{RES}(\text{AF} \downarrow_U)$ , which implies that  $(E \cap U) \in \mathcal{UR}(\text{AF} \downarrow_U, \mathcal{GR})$ . To see that  $(E \cap U) \in \mathcal{MIN}(\mathcal{UR}(\text{AF} \downarrow_U, \mathcal{GR}))$  suppose by contradiction that  $\exists \text{AF}^\sim \in \mathcal{RES}(\text{AF} \downarrow_U) : \text{GE}(\text{AF}^\sim) \subsetneq (E \cap U)$ . To see that this is impossible, consider now an argumentation framework  $\text{AF}'' = \langle \mathcal{A}, \rightarrow'' \rangle$  such that  $\text{AF}'' \downarrow_U = \text{AF}^\sim$ ,  $\text{AF}'' \downarrow_{\mathcal{A} \setminus U} = \text{AF}' \downarrow_{\mathcal{A} \setminus U}$ , and  $\forall (\alpha, \beta) \in \rightarrow : \alpha \in U, \beta \notin U \ (\alpha, \beta) \in \rightarrow''$ . In words,  $\text{AF}''$  is the resolution of  $\text{AF}$  obtained by applying within  $U$  the same substitutions of mutual into unidirectional attacks as in  $\text{AF}^\sim$ , and the same substitutions as in  $\text{AF}'$  outside  $U$ . Note in particular that since  $U$  is unattacked, any attack involving an element  $\alpha \in U$  and an element  $\beta \notin U$  is unidirectional and has the form  $\alpha \rightarrow \beta$ , therefore the same attack is necessarily present in  $\text{AF}'$  (and in any other resolution of  $\text{AF}$ ). We will show that  $\text{GE}(\text{AF}'') \subsetneq \text{GE}(\text{AF}') = E$  which contradicts the hypothesis that  $E \in \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{GR}))$ . First note that, by directionality of grounded semantics we have that  $\text{GE}(\text{AF}'') \cap U = \text{GE}(\text{AF}'' \downarrow_U) = \text{GE}(\text{AF}^\sim) \subsetneq (E \cap U) = \text{GE}(\text{AF}' \downarrow_U) = \text{GE}(\text{AF}') \cap U$ , yielding in particular

$$\text{GE}(\text{AF}'') \cap U \subsetneq \text{GE}(\text{AF}') \cap U \tag{7}$$

It is then sufficient to show that  $\text{GE}(\text{AF}'') \subseteq \text{GE}(\text{AF}')$ . To this purpose, recall from [4] that for any finite  $\text{AF}$ ,  $\text{GE}(\text{AF}) = \bigcup_{i \geq 1} F_{\text{AF}}^i(\emptyset)$ , where  $F_{\text{AF}}^1(\emptyset) = F_{\text{AF}}(\emptyset)$  and  $\forall i > 1$   $F_{\text{AF}}^i(\emptyset) = F_{\text{AF}}(F_{\text{AF}}^{i-1}(\emptyset))$ . It is easy to see that  $F_{\text{AF}''}^1(\emptyset) \subseteq F_{\text{AF}'}^1(\emptyset)$ . In fact for any  $\text{AF}$ ,  $F_{\text{AF}}^1(\emptyset) = \text{IN}(\text{AF})$  (see Definition 2). By construction,  $\text{IN}(\text{AF}'') \cap (\mathcal{A} \setminus U) = \text{IN}(\text{AF}') \cap (\mathcal{A} \setminus U)$  while (7) entails that  $\text{IN}(\text{AF}'') \cap U \subseteq \text{IN}(\text{AF}') \cap U$ . Having shown that  $F_{\text{AF}''}^1(\emptyset) \subseteq F_{\text{AF}'}^1(\emptyset) \subseteq \text{GE}(\text{AF}')$  let us prove by induction that if  $F_{\text{AF}''}^i(\emptyset) \subseteq \text{GE}(\text{AF}')$  then  $F_{\text{AF}''}^{i+1}(\emptyset) \subseteq \text{GE}(\text{AF}')$ . Consider any argument  $\alpha$  in  $F_{\text{AF}''}^{i+1}(\emptyset) \setminus F_{\text{AF}''}^i(\emptyset)$ . If  $\alpha \in U$  then from (7) we directly have that  $\alpha \in \text{GE}(\text{AF}')$ . If  $\alpha \notin U$  then by definition of  $F_{\text{AF}''}^i(\emptyset)$  it holds that  $\forall \beta : \beta \rightarrow \alpha$  in  $\text{AF}'' \ \exists \gamma \in F_{\text{AF}''}^i(\emptyset) : \gamma \rightarrow \beta$  in  $\text{AF}''$ , and by construction of  $\text{AF}''$  the defeaters of  $\alpha$  in  $\text{AF}''$  are precisely those in  $\text{AF}'$ , therefore  $\forall \beta : \beta \rightarrow \alpha$  in  $\text{AF}' \ \exists \gamma \in F_{\text{AF}''}^i(\emptyset) : \gamma \rightarrow \beta$  in  $\text{AF}''$ . Now, by the inductive hypothesis  $\gamma \in \text{GE}(\text{AF}')$ , and since both  $\text{AF}'$  and  $\text{AF}''$  belong to  $\mathcal{RES}(\text{AF})$  either  $\gamma$  attacks  $\beta$  or  $\beta$  attacks  $\gamma$  in  $\text{AF}'$ , where in the latter case, by the admissibility of  $\text{GE}(\text{AF}')$  there must be another element of  $\text{GE}(\text{AF}')$  attacking  $\beta$  in  $\text{AF}'$ . Summing up, when  $\alpha \notin U$  in any



**Figure 1.** An example showing that  $\mathcal{PR}^*$  is not directional.

case all its defeators in  $\text{AF}'$  are in turn attacked in  $\text{GE}(\text{AF}')$ : since the latter is a complete extension it must be the case that  $\alpha \in \text{GE}(\text{AF}')$ .

Turning to the second part of the proof and letting  $U \in \mathcal{US}(\text{AF})$  and  $F \in \mathcal{ER}^*(\text{AF} \downarrow_U) = \mathcal{MIN}(\mathcal{UR}(\text{AF} \downarrow_U, \mathcal{GR}))$ , we have to show that  $\exists E \in \mathcal{ER}^*(\text{AF}) = \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{GR}))$  such that  $E \cap U = F$ . First consider the set  $\mathcal{H} = \{E \in \mathcal{UR}(\text{AF}, \mathcal{GR}) \mid E \cap U = F\}$ . We know that  $\mathcal{H} \neq \emptyset$ . In fact  $F = \text{GE}(\text{AF}^\sim)$  for some  $\text{AF}^\sim \in \mathcal{RES}(\text{AF} \downarrow_U)$ . By Lemma 2,  $\exists \text{AF}' \in \mathcal{RES}(\text{AF}) : \text{AF}' \downarrow_U = \text{AF}^\sim$ . Then  $F = \text{GE}(\text{AF}' \downarrow_U)$  and, by directionality of  $\mathcal{GR}$ ,  $\text{GE}(\text{AF}') \cap U = F$ , where  $\text{GE}(\text{AF}') \in \mathcal{UR}(\text{AF}, \mathcal{GR})$ . Therefore  $\text{GE}(\text{AF}') \in \mathcal{H}$ . Now, we have to show that  $\mathcal{H} \cap \mathcal{MIN}(\mathcal{UR}(\text{AF}, \mathcal{GR})) \neq \emptyset$ , namely that at least an element of  $\mathcal{H}$  is also a minimal element of  $\mathcal{UR}(\text{AF}, \mathcal{GR})$ . Suppose by contradiction that this is not the case. Since  $\mathcal{H} \subseteq \mathcal{UR}(\text{AF}, \mathcal{GR})$  this entails that  $\forall E \in \mathcal{H} \exists E^* \in (\mathcal{UR}(\text{AF}, \mathcal{GR}) \setminus \mathcal{H})$  such that  $E^* \subsetneq E$ . Now consider  $E^* \cap U$ : since  $E^* \subsetneq E$ , it must be the case that  $(E^* \cap U) \subseteq (E \cap U) = F$ , but since  $E^* \in (\mathcal{UR}(\text{AF}, \mathcal{GR}) \setminus \mathcal{H})$  it can not be the case that  $(E^* \cap U) = F$ , therefore  $(E^* \cap U) \subsetneq F$ . Summing up,  $\exists E^* \in \mathcal{UR}(\text{AF}, \mathcal{GR}) \mid (E^* \cap U) \subsetneq F$ . Since  $E^* \in \mathcal{UR}(\text{AF}, \mathcal{GR})$  there exists  $\text{AF}'' \in \mathcal{RES}(\text{AF}) \mid E^* = \text{GE}(\text{AF}')$ . Since  $U \in \mathcal{US}(\text{AF})$  it also holds that  $U \in \mathcal{US}(\text{AF}'')$  and, by directionality of  $\mathcal{GR}$ ,  $\text{GE}(\text{AF}'') \cap U = \text{GE}(\text{AF}'' \downarrow_U)$ . By Lemma 2,  $\text{AF}'' \downarrow_U \in \mathcal{RES}(\text{AF} \downarrow_U)$ , and therefore  $\text{GE}(\text{AF}'') \cap U \in \mathcal{UR}(\text{AF} \downarrow_U, \mathcal{GR})$  but  $\text{GE}(\text{AF}'') \cap U \subsetneq F$  and this contradicts the hypothesis that  $F \in \mathcal{MIN}(\mathcal{UR}(\text{AF} \downarrow_U, \mathcal{GR}))$ .  $\square$

While the resolution-based version of grounded semantics achieves a full satisfaction of the criteria discussed in Section 2, it can be shown that this is not the case for the resolution-based version of preferred semantics, which is not directional. Consider the argumentation framework  $\text{AF}$  shown in Figure 1 along with its resolutions  $\text{AF}'$  and  $\text{AF}''$  and  $U = \{\alpha, \beta, \gamma\} \in \mathcal{US}(\text{AF})$ . It is easy to see that  $\mathcal{ER}^*(\text{AF} \downarrow_U) = \{\emptyset\}$ . On the other hand, since  $\mathcal{ER}(\text{AF}') = \{\{\alpha, \epsilon, \eta\}, \{\alpha, \theta, \iota, \zeta\}\}$  and  $\mathcal{ER}(\text{AF}'') = \{\{\theta, \iota, \zeta\}\}$ , we have  $\mathcal{ER}^*(\text{AF}) = \{\{\alpha, \epsilon, \eta\}, \{\theta, \iota, \zeta\}\}$ . Therefore  $\{(E \cap U) \mid E \in \mathcal{ER}^*(\text{AF})\} = \{\{\alpha\}, \emptyset\} \neq \mathcal{ER}^*(\text{AF} \downarrow_U) = \{\emptyset\}$ .

## 4. Conclusions

We have shown that the resolution-based version of grounded semantics  $\mathcal{GR}^*$  is able to satisfy all the desirable semantics evaluation criteria introduced in [2] (namely I-maximality, admissibility, reinstatement, weak reinstatement,  $\mathcal{CF}$ -reinstatement, directionality,  $\preceq_{\cap}^E$ -and  $\preceq_W^E$ -skepticism adequacy,  $\preceq_{\cap}^E$ - and  $\preceq_W^E$ -resolution adequacy), while the resolution-based version of preferred semantics  $\mathcal{PR}^*$  fails to satisfy directionality. From the theoretical side, the result about  $\mathcal{GR}^*$  proves that these criteria are not incompatible altogether, which can be regarded as a sort of confirmation of their soundness. Actually, the idea of a parametric family of semantics (called *principle-based*) to check satisfiability of criteria was first introduced in [7], but, differently from  $\mathcal{GR}^*$  considered here, none of the principle-based semantics considered in [7] was able to satisfy all the criteria. Further research on this side may concern exploring the properties of  $\mathcal{GR}^*$  and investigating the resolution-based versions of other literature semantics not considered in this paper. From the application side, we regard as an open question whether  $\mathcal{GR}^*$  can be useful in practice. In this respect, it is worth noting that computing the extensions of  $\mathcal{GR}^*$  on AF requires evaluating the grounded extension (a problem which, as well known, is computationally tractable) of all the argumentation frameworks in the set  $\mathcal{RES}(\text{AF})$  whose cardinality is exponential in the number of mutual attacks in AF. Investigating the existence of algorithms which do not require the explicit consideration of all elements of  $\mathcal{RES}(\text{AF})$  to compute the extensions of  $\mathcal{GR}^*$  (possibly for some specific families of argumentation frameworks) is another interesting direction of future work.

## Acknowledgements

We are indebted to the anonymous referees for their helpful comments.

## References

- [1] M. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.
- [2] P. Baroni and M. Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence (Special issue on Argumentation in A.I.)*, 171(10/15):675–700, 2007.
- [3] S. Modgil. Hierarchical argumentation. In *Proc. of the 10th European Conference on Logics in Artificial Intelligence (JELIA 06)*, pages 319–332, Liverpool, UK, 2006. Springer.
- [4] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
- [5] P. Baroni, M. Giacomin, and G. Guida. Towards a formalization of skepticism in extension-based argumentation semantics. In *Proceedings of the 4th Workshop on Computational Models of Natural Argument (CMNA 2004)*, pages 47–52, Valencia, Spain, 2004.
- [6] P. Baroni and M. Giacomin. Evaluating argumentation semantics with respect to skepticism adequacy. In *Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, pages 329–340, Barcelona, E, 2005.
- [7] P. Baroni and M. Giacomin. On principle-based evaluation, comparison, and design of extension-based argumentation semantics. Technical report, University of Brescia, Italy, 2007.

# A Systematic Classification of Argumentation Frameworks where Semantics Agree

Pietro BARONI<sup>a,1</sup> and Massimiliano GIACOMIN<sup>a</sup>

<sup>a</sup> *Dip. Elettronica per l'Automazione, Univ. of Brescia, Italy*

**Abstract.** The issue of characterizing classes of argumentation frameworks where different semantics agree has been considered in the literature with main focus on the relationships between agreement and topological properties. This paper contributes to this kind of investigation from a complementary perspective, by introducing a systematic classification of agreement classes concerning a comprehensive set of argumentation semantics on the basis of properties of their sets of extensions only. In particular, it is shown that 14 distinct classes out of 120 nominal ones exist, and a complete analysis of their set-theoretical relationships is carried out.

**Keywords.** Argumentation semantics, Argumentation frameworks

## Introduction

The importance of systematic principle-based assessment and comparison of different argumentation semantics is increasingly recognized [1,2], given the variety of approaches available in the literature. While comparisons are often focused on the differences between alternative proposals, it is also interesting to characterize situations where argumentation semantics agree, i.e. exhibit the same behavior in spite of their differences. This can be useful from several viewpoints. On one hand, situations where “most” (or even all) existing semantics agree can be regarded as providing a sort of reference behavior against which further proposals should be confronted. On the other hand, it may be the case that in a specific application domain there are some restrictions on the structure of the argumentation frameworks that need to be considered. It is then surely interesting to know whether these restrictions lead to semantics agreement, since in this case it is clear that evaluations about arguments in that domain are not affected by different choices of argumentation semantics and are, in a sense, universally supported.

This paper provides a contribution to this research direction by analyzing *agreement classes* of argumentation frameworks with respect to a set of seven semantics, representing traditional and more recent literature proposals. We show that there exist 14 distinct agreement classes (out of a potential number of 120) and analyze their set-theoretical relationships. The paper is organized as follows. After recalling the necessary background

---

<sup>1</sup>Corresponding Author: Pietro Baroni, Dip. Elettronica per l'Automazione, Univ. of Brescia, Via Branze 38, 25123 Brescia, Italy. Tel.: +39 030 3715455; Fax: +39 030 380014; E-mail: baroni@ing.unibs.it.

concepts in Section 1, we review in Section 2 the argumentation semantics considered in the paper. The analysis of agreement classes is carried out in Section 3, while Section 4 concludes the paper.

## 1. Background Concepts and Notation

The present work lies in the frame of the general theory of abstract argumentation frameworks proposed by Dung [3].

**Definition 1** *An argumentation framework is a pair  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , where  $\mathcal{A}$  is a set, and  $\rightarrow \subseteq (\mathcal{A} \times \mathcal{A})$  is a binary relation on  $\mathcal{A}$ , called attack relation.*

In the following we will always assume that  $\mathcal{A}$  is finite. We extend to set of arguments the notion of attack as follows: given a set  $E \subseteq \mathcal{A}$ ,  $E \rightarrow \alpha \equiv \exists \beta \in E : \beta \rightarrow \alpha$ ;  $\alpha \rightarrow E \equiv \exists \beta \in E : \alpha \rightarrow \beta$ .

Two particular kinds of elementary argumentation frameworks need to be introduced as they will play some role in the following. The *empty argumentation framework*, denoted as  $\text{AF}_\emptyset$ , is simply defined as  $\text{AF}_\emptyset = \langle \emptyset, \emptyset \rangle$ . Furthermore, an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  is *monadic* if  $|\mathcal{A}| = 1$  and  $\rightarrow = \emptyset$ .

Finally we introduce an operation of composition of argumentation frameworks, denoted as  $\sqcup$ , which will be useful to shorten some notations and definitions. Given two argumentation frameworks  $\text{AF}_1 = \langle \mathcal{A}_1, \rightarrow_1 \rangle$  and  $\text{AF}_2 = \langle \mathcal{A}_2, \rightarrow_2 \rangle$ , such that  $\mathcal{A}_1 \cap \mathcal{A}_2 = \emptyset$ ,  $\text{AF}_1 \sqcup \text{AF}_2 = \langle \mathcal{A}_1 \cup \mathcal{A}_2, \rightarrow_1 \cup \rightarrow_2 \rangle$ .

In Dung's theory, an argumentation semantics is defined by specifying the criteria for deriving, given a generic argumentation framework, the set of all possible extensions, each one representing a set of arguments considered to be acceptable together. Accordingly, a basic requirement for any extension  $E$  is that it is *conflict-free*, namely  $\nexists \alpha, \beta \in E : \alpha \rightarrow \beta$ . All argumentation semantics proposed in the literature satisfy this fundamental *conflict-free property*.

Given a generic argumentation semantics  $\mathcal{S}$ , the set of extensions prescribed by  $\mathcal{S}$  for a given argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  is denoted as  $\mathcal{E}_{\mathcal{S}}(\text{AF})$ . If it holds that  $\forall \text{AF} \ |\mathcal{E}_{\mathcal{S}}(\text{AF})| = 1$ , then the semantics  $\mathcal{S}$  is said to follow the *unique-status approach*, otherwise it is said to follow the *multiple-status approach*. A relevant question concerns the existence of extensions. While the definition of most semantics ensures that there is at least one extension for any argumentation framework, it is well-known that this is not always the case (in particular for stable semantics). Formally, for a generic semantics  $\mathcal{S}$  we will denote as  $\mathcal{D}_{\mathcal{S}}$  the set of argumentation frameworks where  $\mathcal{S}$  admits at least one extension, namely  $\mathcal{D}_{\mathcal{S}} = \{\text{AF} : \mathcal{E}_{\mathcal{S}}(\text{AF}) \neq \emptyset\}$ .

In the following, whenever we will discuss the properties of agreement of a semantics  $\mathcal{S}$  we will implicitly refer only to argumentation frameworks belonging to  $\mathcal{D}_{\mathcal{S}}$  if not differently specified. Note, in particular, that if  $\text{AF}_\emptyset \in \mathcal{D}_{\mathcal{S}}$  then necessarily  $\mathcal{E}_{\mathcal{S}}(\text{AF}_\emptyset) = \{\emptyset\}$ . We can now formally define the notion of agreement: we will say that two semantics  $\mathcal{S}_1$  and  $\mathcal{S}_2$  are in agreement on an argumentation framework  $\text{AF} \in \mathcal{D}_{\mathcal{S}_1} \cap \mathcal{D}_{\mathcal{S}_2}$  if  $\mathcal{E}_{\mathcal{S}_1}(\text{AF}) = \mathcal{E}_{\mathcal{S}_2}(\text{AF})$ . Letting  $\mathbb{S}$  a set of argumentation semantics, the set of argumentation frameworks where all semantics included in  $\mathbb{S}$  are in agreement will be denoted as  $\mathcal{AGR}(\mathbb{S})$ . As stated above, if a semantics  $\mathcal{S} \in \mathbb{S}$  is not universally defined then  $\mathcal{AGR}(\mathbb{S}) \subseteq \mathcal{D}_{\mathcal{S}}$ . It can also be noted that, in general, it may be the case that

$\mathcal{AGR}(\mathbb{S}_1) = \mathcal{AGR}(\mathbb{S}_2)$  for different sets of semantics  $\mathbb{S}_1$  and  $\mathbb{S}_2$ , and it is obviously the case that  $\mathbb{S}_1 \subseteq \mathbb{S}_2 \Rightarrow \mathcal{AGR}(\mathbb{S}_2) \subseteq \mathcal{AGR}(\mathbb{S}_1)$ .

## 2. A Review of Extension-Based Argumentation Semantics

To make the paper self-contained, in this section we quickly review the definition of the argumentation semantics considered in our analysis, namely stable, complete, grounded, preferred, *CF2*, semi-stable and ideal semantics.

### 2.1. Traditional Semantics

Stable semantics relies on the idea that an extension should not only be internally consistent but also able to reject the arguments that are outside the extension. This reasoning leads to the notion of stable extension [3].

**Definition 2** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , a set  $E \subseteq \mathcal{A}$  is a stable extension of AF if and only if  $E$  is conflict-free and  $\forall \alpha \in \mathcal{A} : \alpha \notin E, E \rightarrow \alpha$ .*

Stable semantics will be denoted as  $\mathcal{ST}$ , and, accordingly, the set of all the stable extensions of AF will be denoted as  $\mathcal{E}_{\mathcal{ST}}(\text{AF})$ . Stable semantics suffers from a significant limitation since there are argumentation frameworks where no extension complying with Definition 2 exists. No other semantics considered in this paper is affected by this problem. The requirement that an extension should attack all other external arguments can be relaxed by imposing that an extension is simply able to defend itself from external attacks. This is at the basis of the notions of acceptable argument, admissible set and characteristic function [3].

**Definition 3** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , an argument  $\alpha \in \mathcal{A}$  is acceptable with respect to a set  $E \subseteq \mathcal{A}$  if and only if  $\forall \beta \in \mathcal{A} : \beta \rightarrow \alpha, E \rightarrow \beta$ . A set  $E \subseteq \mathcal{A}$  is admissible if and only if  $E$  is conflict-free and  $\forall \beta \in \mathcal{A} : \beta \rightarrow E, E \rightarrow \beta$ . The set of the admissible sets of AF is denoted as  $\mathcal{AS}(\text{AF})$ . The function  $F_{\text{AF}} : 2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$  which, given a set  $E \subseteq \mathcal{A}$ , returns the set of the acceptable arguments with respect to  $E$ , is called the characteristic function of AF.*

Building on these concepts, the notion of complete extension can be introduced, which plays a key role in Dung's theory, since all semantics encompassed by his framework select their extensions among the complete ones.

**Definition 4** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , a set  $E \subseteq \mathcal{A}$  is a complete extension if and only if  $E$  is admissible and every argument of  $\mathcal{A}$  which is acceptable with respect to  $E$  belongs to  $E$ , i.e.  $E \in \mathcal{AS}(\text{AF}) \wedge \forall \alpha \in F_{\text{AF}}(E) \alpha \in E$ .*

The notion of complete extension is not associated to a notion of *complete semantics* in [3], but rather represents an intermediate step towards the definition of grounded and preferred semantics. However, the term complete semantics has subsequently gained acceptance in the literature and will be used in the present analysis to refer to the properties of the set of complete extensions. Complete semantics will be denoted as  $\mathcal{CO}$ .

The well-known grounded semantics, denoted as  $\mathcal{GR}$ , belongs to the unique-status approach and its unique extension, denoted as  $\text{GE}(\text{AF})$ , can be defined as the least fixed point of the characteristic function.

**Definition 5** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , the grounded extension of  $\text{AF}$ , denoted as  $\text{GE}(\text{AF})$ , is the least fixed point (with respect to set inclusion) of  $F_{\text{AF}}$ .*

Preferred semantics, denoted as  $\mathcal{PR}$ , is obtained by simply requiring the property of maximality along with admissibility.

**Definition 6** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , a set  $E \subseteq \mathcal{A}$  is a preferred extension of  $\text{AF}$  if and only if it is a maximal (with respect to set inclusion) admissible set, i.e. a maximal element of  $\mathcal{AS}(\text{AF})$ .*

## 2.2. CF2 Semantics

*CF2 semantics*, first introduced in [4], is a SCC-recursive semantics [5] which features the distinctive property of treating in a “symmetric” way odd- and even-length cycles while belonging to the multiple-status approach. SCC-recursiveness is related to the graph-theoretical notion of *strongly connected components* (SCCs) of  $\text{AF}$ , namely the equivalence classes of nodes under the relation of mutual reachability, denoted as  $\text{SCCS}_{\text{AF}}$ . Due to space limitations, we can not examine in detail the definition of *CF2* semantics: the interested reader may refer to [4] and [5].

**Definition 7** *Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$ , a set  $E \subseteq \mathcal{A}$  is an extension of *CF2* semantics iff*

- $E \in \mathcal{MCF}_{\text{AF}}$  if  $|\text{SCCS}_{\text{AF}}| = 1$
- $\forall S \in \text{SCCS}_{\text{AF}} (E \cap S) \in \mathcal{E}_{\text{CF2}}(\text{AF} \downarrow_{\text{UP}_{\text{AF}}(S, E)})$  otherwise

where  $\mathcal{MCF}_{\text{AF}}$  denotes the set of maximal conflict-free sets of  $\text{AF}$ , and, for any set  $S \subseteq \mathcal{A}$ ,  $\text{AF} \downarrow_S$  denotes the restriction of  $\text{AF}$  to  $S$ , namely  $\text{AF} \downarrow_S = \langle S, \rightarrow \cap (S \times S) \rangle$ , and  $\text{UP}_{\text{AF}}(S, E) = \{\alpha \in S \mid \#\beta \in E : \beta \notin S, \beta \rightarrow \alpha\}$ .

As shown by the following lemma, *CF2* semantics can be roughly regarded as selecting its extensions among the maximal conflict free sets of  $\text{AF}$ , on the basis of some topological requirements related to the decomposition of  $\text{AF}$  into strongly connected components.

**Lemma 1** (*Lemma 2 of [6]*) *For any argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$   $\mathcal{E}_{\text{CF2}}(\text{AF}) \subseteq \mathcal{MCF}_{\text{AF}}$ , where  $\mathcal{MCF}_{\text{AF}}$  denotes the set of all the maximal (with respect to set inclusion) conflict-free sets of  $\text{AF}$ .*

## 2.3. Semi-stable Semantics

Semi-stable semantics [7], denoted in the following as  $\mathcal{SST}$ , aims at guaranteeing the existence of extensions in any case (differently from stable semantics) while coinciding with stable semantics (differently from preferred semantics) when stable extensions exist. The definition of extensions satisfying these desiderata is ingeniously simple.

**Definition 8** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  a set  $E \subseteq \mathcal{A}$  is a semi-stable extension if and only if  $E$  is a complete extension such that  $E \cup \{\alpha \in \mathcal{A} \mid E \rightarrow \alpha\}$  is maximal with respect to set inclusion.

We also recall an important property to be used in the following: semi-stable semantics is always in agreement with stable semantics if stable extensions exist.

**Proposition 1** (Theorem 3 of [7]) If  $\text{AF} \in \mathcal{D}_{ST}$  then  $\mathcal{E}_{SST}(\text{AF}) = \mathcal{E}_{ST}(\text{AF})$ .

#### 2.4. Ideal Semantics

Ideal semantics [8] provides an alternative unique-status approach which is less sceptical than grounded semantics, i.e. for any argumentation framework the (unique) ideal extension is a (sometimes strict) superset of the grounded extension.

**Definition 9** Given an argumentation framework  $\text{AF} = \langle \mathcal{A}, \rightarrow \rangle$  a set  $E \subseteq \mathcal{A}$  is ideal if and only if  $E$  is admissible and  $\forall P \in \mathcal{E}_{PR}(\text{AF}) E \subseteq P$ . The ideal extension is the maximal (with respect to set inclusion) ideal set.

We will use the symbol  $ID$  to refer to the ideal semantics, and the ideal extension of an argumentation framework  $\text{AF}$  will be denoted as  $ID(\text{AF})$ .

#### 2.5. Properties and Relationships

In the following we will often use several known results about the semantics listed above: they are quickly recalled here for the reader's convenience. The I-maximality property for a semantics  $\mathcal{S}$  states that no extension is a proper subset of another one.

**Definition 10** A set of extensions  $\mathcal{E}$  is I-maximal iff  $\forall E_1, E_2 \in \mathcal{E}$ , if  $E_1 \subseteq E_2$  then  $E_1 = E_2$ . A semantics  $\mathcal{S}$  is I-maximal if and only if  $\forall \text{AF}, \mathcal{E}_{\mathcal{S}}(\text{AF})$  is I-maximal.

It is proved in [2] that all semantics considered in this paper except  $\mathcal{CO}$  are I-maximal.

It is proved in [3,5] that all multiple-status semantics considered in this paper satisfy the following inclusion relationship with respect to grounded semantics:  $\forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF}) GE(\text{AF}) \subseteq E$  with  $\mathcal{S} \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{CF2}\}$ . Moreover it is well known [3] that  $GE(\text{AF})$  is the least complete extension, thus in particular  $GE(\text{AF}) \in \mathcal{E}_{\mathcal{CO}}(\text{AF})$ . As to inclusion between sets of extensions, a classical result [3] states that  $\mathcal{E}_{ST}(\text{AF}) \subseteq \mathcal{E}_{PR}(\text{AF}) \subseteq \mathcal{E}_{CO}(\text{AF})$ . Moreover it is proved in [7] that  $\mathcal{E}_{SST}(\text{AF}) \subseteq \mathcal{E}_{PR}(\text{AF})$ .

Since complete extensions are admissible and preferred extensions are maximal admissible sets, it follows that  $\forall E \in \mathcal{E}_{CO}(\text{AF}) \exists E' \in \mathcal{E}_{PR}(\text{AF}) : E \subseteq E'$ . A similar relation holds between preferred and  $\mathcal{CF2}$  semantics [4]:  $\forall E \in \mathcal{E}_{PR}(\text{AF}) \exists E' \in \mathcal{E}_{CF2}(\text{AF}) : E \subseteq E'$ .

### 3. Agreement Classes of Argumentation Frameworks

We focus on the set of semantics  $\Omega = \{\mathcal{GR}, \mathcal{ID}, \mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{CF2}, \mathcal{SST}\}$ , as stated in previous section. Considering all subsets  $\mathbb{S}$  of  $\Omega$  such that  $|\mathbb{S}| \geq 2$  gives rise, in principle,

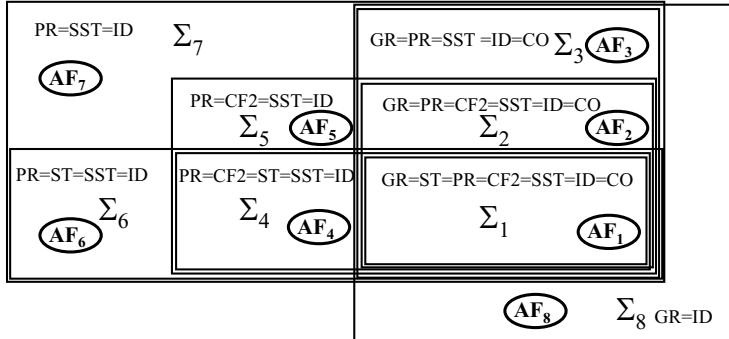


Figure 1. Venn diagram of unique-status agreement classes.

to 120 classes  $\mathcal{AGR}(\mathbb{S})$  to be evaluated. The main result of the paper consists in showing that there are only 14 distinct agreement classes. Note that  $\mathcal{AGR}(\mathbb{S}) \neq \emptyset$  for any set of semantics  $\mathbb{S}$  since  $\text{AF}_\emptyset \in \mathcal{AGR}(\mathbb{S})$  for all semantics belonging to  $\Omega$ . Moreover, it can be noted that all semantics considered in this paper (and probably any other reasonable argumentation semantics) are in agreement also on monadic argumentation frameworks.

We will start by analyzing in subsection 3.1 unique-status agreement, namely the classes  $\mathcal{AGR}(\mathbb{S})$  where  $\mathbb{S}$  includes  $\mathcal{GR}$  or  $\mathcal{ID}$ , and we will then examine agreement between multiple-status semantics in subsection 3.2.

### 3.1. Unique-status Agreement

The Venn diagram concerning unique-status agreement classes is shown in Figure 1 where rectangular boxes represent classes of agreement and small ellipses represent single argumentation frameworks to be used as specific examples. The diagram will be illustrated in two main steps: first, we will show that the set-theoretical relationships between the agreement classes  $\Sigma_1, \dots, \Sigma_8$  depicted in Figure 1 actually hold, then we will prove that the classes  $\Sigma_1, \dots, \Sigma_8$  are the only meaningful ones in the context of unique-status agreement.

As to the first step, we proceed by following the partial order induced by inclusion (namely if  $\Sigma_i \subsetneq \Sigma_j$  then  $j > i$ ). While introducing each class  $\Sigma_i$  it will be necessary:

1. to identify which classes  $\Sigma_k$ ,  $k < i$ , are included in  $\Sigma_i$ ;
2. for each of these classes  $\Sigma_k$  to show that  $\Sigma_i \setminus \Sigma_k \neq \emptyset$ ;
3. for any  $\Sigma_h$  such that  $h < i$  and  $\Sigma_h \not\subseteq \Sigma_i$  to examine  $\Sigma_i \cap \Sigma_h$ .

Point 1 will not be stressed since inclusion relationships can be directly derived from the inclusion of the relevant sets of semantics ( $\mathbb{S}_1 \subseteq \mathbb{S}_2 \Rightarrow \mathcal{AGR}(\mathbb{S}_2) \subseteq \mathcal{AGR}(\mathbb{S}_1)$ ). As to point 2, examples of argumentation frameworks belonging to non-empty set differences will be given to prove that inclusion relationships are strict, while point 3 will be dealt with case by case.

Let us start from  $\mathcal{AGR}(\{\mathcal{GR}, \mathcal{ID}, \mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{CF2}, \mathcal{SST}\})$ , denoted as  $\Sigma_1$ . This is the class of argumentation frameworks where all semantics show a uniform single status behavior in agreement with grounded semantics.  $\Sigma_1$  includes, for instance, monadic argumentation frameworks like  $\text{AF}_1 = \langle \{\alpha\}, \emptyset \rangle$ .

$\text{AGR}(\{\mathcal{GR}, \mathcal{ID}, \mathcal{CO}, \mathcal{PR}, \mathcal{CF2}, \mathcal{SST}\})$ , denoted as  $\Sigma_2$ , corresponds to a uniform single status behavior in agreement with grounded semantics by all but stable semantics. As shown in Figure 1,  $\Sigma_2$  strictly includes  $\Sigma_1$  since  $(\Sigma_2 \setminus \Sigma_1)$  includes in particular  $\text{AF}_2 = \langle \{\alpha, \beta\}, \{(\beta, \beta)\} \rangle$ .  $\Sigma_3 \triangleq \text{AGR}(\{\mathcal{GR}, \mathcal{ID}, \mathcal{CO}, \mathcal{PR}, \mathcal{SST}\})$  is the last class in Figure 1 concerning agreement with grounded semantics.  $(\Sigma_3 \setminus \Sigma_2) \neq \emptyset$  since it includes for instance  $\text{AF}_3 = \langle \{\alpha, \beta, \gamma\}, \{(\alpha, \beta), (\beta, \gamma), (\gamma, \alpha)\} \rangle$ . It is now worth noting that  $\Sigma_3 \cap \mathcal{D}_{ST} = \Sigma_2 \cap \mathcal{D}_{ST} = \Sigma_1$ . In fact, by Proposition 1, whenever  $\text{AF} \in \mathcal{D}_{ST}$  it must be the case that  $\text{AF} \in \text{AGR}(\mathbb{S})$  where  $\mathbb{S}$  includes both  $ST$  and  $SST$ .

On the left of  $\Sigma_1$ ,  $\Sigma_2$  and  $\Sigma_3$  the diagram of Figure 1 shows four classes where several multiple-status semantics exhibit a unique-status behavior in agreement with ideal semantics, but not necessarily also with grounded semantics. The smallest of these classes is  $\Sigma_4 \triangleq \text{AGR}(\{\mathcal{ID}, \mathcal{CF2}, \mathcal{ST}, \mathcal{PR}, \mathcal{SST}\})$ .  $(\Sigma_4 \setminus \Sigma_1) \neq \emptyset$  since it includes  $\text{AF}_4 = \langle \{\alpha, \beta\}, \{(\alpha, \beta), (\beta, \alpha), (\beta, \beta)\} \rangle$ . Moreover, since  $\Sigma_4 \subset \mathcal{D}_{ST}$ , it is the case that  $\Sigma_4 \cap (\Sigma_3 \setminus \Sigma_1) = \emptyset$ . Not requiring agreement with stable semantics leads to  $\Sigma_5 \triangleq \text{AGR}(\{\mathcal{ID}, \mathcal{CF2}, \mathcal{PR}, \mathcal{SST}\})$ .  $(\Sigma_5 \setminus (\Sigma_4 \cup \Sigma_2)) \neq \emptyset$ , since it includes  $\text{AF}_5 = \langle \{\alpha, \beta, \gamma\}, \{(\alpha, \beta), (\beta, \alpha), (\beta, \beta), (\gamma, \gamma)\} \rangle$ . Notice also that  $\Sigma_5 \cap (\Sigma_3 \setminus \Sigma_2) = \emptyset$  since if  $\text{AF} \in \Sigma_5 \cap \Sigma_3$  then, by definition of these classes, it also holds  $\text{AF} \in \Sigma_2$ . It is also clear that  $\Sigma_5 \cap \mathcal{D}_{ST} = \Sigma_4$ . Not requiring agreement with  $CF2$  semantics leads to  $\Sigma_6 \triangleq \text{AGR}(\{\mathcal{ID}, \mathcal{ST}, \mathcal{PR}, \mathcal{SST}\})$ .  $(\Sigma_6 \setminus \Sigma_4) \neq \emptyset$  since it includes  $\text{AF}_6 = \langle \{\alpha, \beta, \gamma\}, \{(\alpha, \beta), (\alpha, \gamma), (\beta, \gamma), (\gamma, \alpha)\} \rangle$ . Again, since  $\Sigma_6 \subset \mathcal{D}_{ST}$  it holds  $\Sigma_6 \cap (\Sigma_3 \setminus \Sigma_1) = \emptyset$  and  $\Sigma_6 \cap (\Sigma_5 \setminus \Sigma_4) = \emptyset$ . Finally, excluding both stable and  $CF2$  semantics from the required agreement corresponds to  $\Sigma_7 \triangleq \text{AGR}(\{\mathcal{ID}, \mathcal{PR}, \mathcal{SST}\})$ .  $(\Sigma_7 \setminus (\Sigma_6 \cup \Sigma_5 \cup \Sigma_3)) \neq \emptyset$ , since it includes  $\text{AF}_7 = \langle \{\alpha, \beta, \gamma, \delta, \epsilon\}, \{(\alpha, \beta), (\beta, \alpha), (\beta, \beta), (\gamma, \delta), (\delta, \epsilon), (\epsilon, \gamma)\} \rangle$ .

The last class concerning unique-status agreement is  $\Sigma_8 \triangleq \text{AGR}(\{\mathcal{ID}, \mathcal{GR}\})$ . By definition of the relevant sets of semantics (and since no other distinct agreement classes exist, as it will be shown in a while) it holds that  $\Sigma_8 \cap \Sigma_7 = \Sigma_3$ , moreover it is easy to see that  $\Sigma_8 \setminus \Sigma_3 \neq \emptyset$ , since it includes  $\text{AF}_8 = \langle \{\alpha, \beta\}, \{(\alpha, \beta), (\beta, \alpha)\} \rangle$ .

We have now to show that no other classes than  $\Sigma_1, \dots, \Sigma_8$  are meaningful in the context of unique-status agreement. A sequence of preliminary lemmata is needed.

**Lemma 2** If  $|\mathcal{E}_{\mathcal{PR}}(\text{AF})| = 1$ , then 1)  $\mathcal{E}_{\mathcal{PR}}(\text{AF}) = \mathcal{E}_{\mathcal{ID}}(\text{AF})$ ; 2)  $\mathcal{E}_{\mathcal{PR}}(\text{AF}) = \mathcal{E}_{SST}(\text{AF})$ ; 3) if  $\text{AF} \in \mathcal{D}_{ST}$ ,  $\mathcal{E}_{\mathcal{PR}}(\text{AF}) = \mathcal{E}_{ST}(\text{AF})$ .

*Proof.* 1) immediate from definition of  $\mathcal{ID}$ ; 2) follows from  $\emptyset \neq \mathcal{E}_{SST}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{PR}}(\text{AF})$ ; 3) follows from  $\emptyset \neq \mathcal{E}_{ST}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{PR}}(\text{AF})$ .  $\square$

**Lemma 3** If  $|\mathcal{E}_{\mathcal{CO}}(\text{AF})| = 1$ , then  $\mathcal{E}_{\mathcal{CO}}(\text{AF}) = \mathcal{E}_{\mathcal{GR}}(\text{AF}) = \mathcal{E}_{\mathcal{PR}}(\text{AF})$ .

*Proof.* The conclusion follows from  $\emptyset \neq \mathcal{E}_{\mathcal{PR}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{CO}}(\text{AF})$  and  $\text{GE}(\text{AF}) \in \mathcal{E}_{\mathcal{CO}}(\text{AF})$ .  $\square$

**Lemma 4** Let  $\mathcal{S} \in \{\mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{CF2}\}$ , if  $\text{GE}(\text{AF}) \in \mathcal{E}_{\mathcal{S}}(\text{AF})$  then  $\mathcal{E}_{\mathcal{S}}(\text{AF}) = \{\text{GE}(\text{AF})\}$ .

*Proof.* It is known that for any semantics  $\mathcal{S} \in \{\mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{CF2}\}$  it holds that  $\forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF}) \text{ GE}(\text{AF}) \subseteq E$ . By the hypothesis,  $\exists E^* \in \mathcal{E}_{\mathcal{S}}(\text{AF}), E^* = \text{GE}(\text{AF})$ . Then  $\forall E \in \mathcal{E}_{\mathcal{S}}(\text{AF}) E^* \subseteq E$ . The conclusion then trivially follows from the I-maximality property of any semantics  $\mathcal{S} \in \{\mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{CF2}\}$ .  $\square$

**Lemma 5** If  $\mathcal{E}_{\mathcal{PR}}(\text{AF}) = \{\text{GE}(\text{AF})\}$  then  $\mathcal{E}_{\mathcal{CO}}(\text{AF}) = \{\text{GE}(\text{AF})\}$ .

*Proof.* The conclusion follows from the fact that  $\forall E \in \mathcal{E}_{\mathcal{CO}}(\text{AF}) \text{ GE}(\text{AF}) \subseteq E$  and  $\forall E \in \mathcal{E}_{\mathcal{CO}}(\text{AF}) \exists E' \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) : E \subseteq E'$ .  $\square$

**Lemma 6** If  $\mathcal{E}_{\mathcal{ST}}(\text{AF}) = \{\text{GE}(\text{AF})\}$  then  $\mathcal{E}_{\mathcal{CF}2}(\text{AF}) = \{\text{GE}(\text{AF})\}$ .

*Proof.* It is known that  $\forall E \in \mathcal{E}_{\mathcal{CF}2}(\text{AF}) \text{ GE}(\text{AF}) \subseteq E$ . By the hypothesis,  $\forall \alpha \notin \text{GE}(\text{AF}) \text{ GE}(\text{AF}) \rightarrow \alpha$ , and since any  $E \in \mathcal{E}_{\mathcal{CF}2}(\text{AF})$  is conflict-free it must be the case that  $E = \text{GE}(\text{AF})$ .  $\square$

**Lemma 7** If  $\mathcal{E}_{\mathcal{CF}2}(\text{AF}) = \mathcal{E}_{\mathcal{GR}}(\text{AF}) = \{\text{GE}(\text{AF})\}$  then  $\mathcal{E}_{\mathcal{CF}2}(\text{AF}) = \mathcal{E}_{\mathcal{PR}}(\text{AF})$ .

*Proof.* It is known that  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \text{ GE}(\text{AF}) \subseteq E$  (in this case  $\text{GE}(\text{AF})$  coincides with the unique  $\mathcal{CF}2$ -extension) and also that  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \exists E' \in \mathcal{E}_{\mathcal{CF}2}(\text{AF}) : E \subseteq E'$ . Then  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \text{ GE}(\text{AF}) \subseteq E \subseteq \text{GE}(\text{AF})$ , and the conclusion follows.  $\square$

**Lemma 8** If  $\mathcal{E}_{\mathcal{CF}2}(\text{AF}) = \mathcal{E}_{\mathcal{ID}}(\text{AF}) = \{\text{ID}(\text{AF})\}$  then  $\mathcal{E}_{\mathcal{CF}2}(\text{AF}) = \mathcal{E}_{\mathcal{PR}}(\text{AF})$ .

*Proof.* By definition of ideal semantics  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \text{ ID}(\text{AF}) \subseteq E$ , while it is known that  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \exists E' \in \mathcal{E}_{\mathcal{CF}2}(\text{AF}) : E \subseteq E'$ . Then  $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \text{ ID}(\text{AF}) \subseteq E \subseteq \text{ID}(\text{AF})$ , yielding the desired conclusion.  $\square$

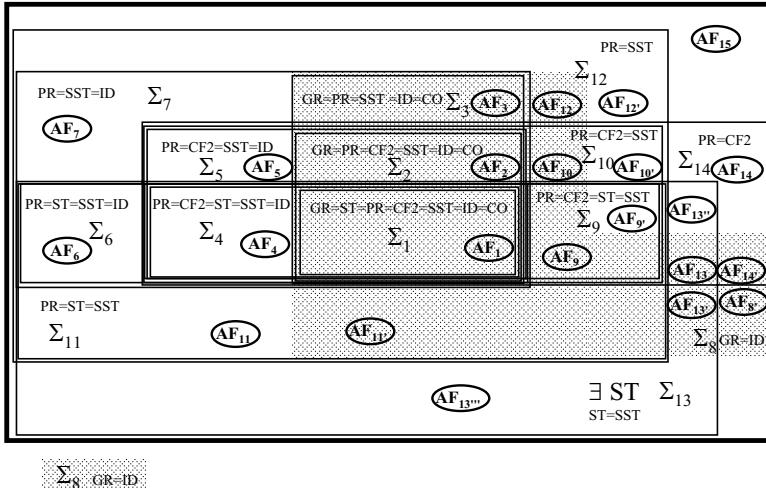
**Lemma 9** If  $\mathcal{E}_{\mathcal{SST}}(\text{AF}) = \mathcal{E}_{\mathcal{ID}}(\text{AF}) = \{\text{ID}(\text{AF})\}$  then  $\mathcal{E}_{\mathcal{SST}}(\text{AF}) = \mathcal{E}_{\mathcal{PR}}(\text{AF})$ .

*Proof.* Since  $\mathcal{E}_{\mathcal{SST}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{PR}}(\text{AF})$ , the hypothesis entails  $\text{ID}(\text{AF}) \in \mathcal{E}_{\mathcal{PR}}(\text{AF})$ . Since by definition of  $\mathcal{ID}$   $\forall E \in \mathcal{E}_{\mathcal{PR}}(\text{AF}) \text{ ID}(\text{AF}) \subseteq E$ , by I-maximality of  $\mathcal{PR}$  we have  $\mathcal{E}_{\mathcal{PR}}(\text{AF}) = \{\text{ID}(\text{AF})\}$ .  $\square$

Having completed the required preliminary results, let us now orderly show that any agreement class  $\mathcal{AGR}(\mathbb{S})$  where  $\mathbb{S}$  includes either  $\mathcal{GR}$  or  $\mathcal{ID}$  coincides with one of the classes  $\Sigma_1, \dots, \Sigma_8$ .

First, we show that  $\Sigma_1$ ,  $\Sigma_2$  and  $\Sigma_3$  are the only classes where  $\mathcal{CO}$  appears. In fact, if  $\{\mathcal{GR}, \mathcal{CO}\} \subseteq \mathbb{S}$  or  $\{\mathcal{ID}, \mathcal{CO}\} \subseteq \mathbb{S}$ , Lemma 3 applies and then also the hypothesis of Lemma 2 is satisfied, which entails  $\mathcal{AGR}(\mathbb{S}) \subseteq \Sigma_3$ . On the other hand, if  $\mathcal{S} \in \{\mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{CF}2\}$  and  $\{\mathcal{S}, \mathcal{CO}\} \subseteq \mathbb{S}$  then since  $\text{GE}(\text{AF}) \in \mathcal{E}_{\mathcal{CO}}(\text{AF})$  the hypothesis of Lemma 4 holds, which implies  $\mathcal{E}_{\mathcal{S}}(\text{AF}) = \{\text{GE}(\text{AF})\} = \mathcal{E}_{\mathcal{CO}}(\text{AF})$ . As in the previous case, this leads to  $\mathcal{AGR}(\mathbb{S}) \subseteq \Sigma_3$ . It remains only to be seen that  $\mathcal{AGR}(\{\mathcal{GR}, \mathcal{PR}, \mathcal{ST}, \mathcal{SST}, \mathcal{ID}, \mathcal{CO}\})$  is not a distinct agreement class: this directly follows from the more general fact that if  $\{\mathcal{GR}, \mathcal{ST}\} \subseteq \mathbb{S}$  then  $\mathcal{AGR}(\mathbb{S}) = \Sigma_1$ . First note that, by Lemma 6, also  $\mathcal{CF}2$  is in agreement. Moreover, recalling that  $\mathcal{E}_{\mathcal{ST}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{SST}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{PR}}(\text{AF})$ , if  $\mathcal{GR}$  and  $\mathcal{ST}$  are in agreement on AF then the hypothesis of Lemma 4 holds for  $\mathcal{S} = \mathcal{PR}$  and  $\mathcal{S} = \mathcal{SST}$ . As a consequence also Lemmata 2 and 5 apply, yielding  $\mathcal{AGR}(\mathbb{S}) = \Sigma_1$ . In virtue of these results,  $\mathcal{CO}$  will not need to be considered any more in the part of the paper concerning multiple-status agreement.

Let us now focus on agreement classes including grounded semantics. We have already seen that if  $\{\mathcal{GR}, \mathcal{ST}\} \subseteq \mathbb{S}$  then  $\mathcal{AGR}(\mathbb{S}) = \Sigma_1$ . Consider now the case where  $\{\mathcal{GR}, \mathcal{CF}2\} \subseteq \mathbb{S} \wedge \mathcal{ST} \notin \mathbb{S}$ : applying Lemma 7 and then Lemmata 2 and 5 it follows directly  $\mathcal{AGR}(\mathbb{S}) = \Sigma_2$ . We can now examine the case  $\{\mathcal{GR}, \mathcal{SST}\} \subseteq \mathbb{S} \wedge (\{\mathcal{ST}, \mathcal{CF}2\} \cap \mathbb{S}) = \emptyset$ : since  $\mathcal{E}_{\mathcal{SST}}(\text{AF}) \subseteq \mathcal{E}_{\mathcal{PR}}(\text{AF})$ , Lemma 4 applies with



**Figure 2.** Venn diagram of agreement classes.

$\mathcal{S} = \mathcal{PR}$ , then from Lemmata 2 and 5 we have  $\mathcal{AGR}(\mathbb{S}) = \Sigma_3$ . By Lemma 2, the case  $\{\mathcal{GR}, \mathcal{PR}\} \subseteq \mathbb{S}$  can be reduced to the cases examined previously. Finally, the only remaining agreement class involving grounded semantics is  $\mathcal{AGR}(\{\mathcal{GR}, \mathcal{ID}\}) = \Sigma_8$ .

Let us now turn to agreement classes involving ideal but not grounded (and complete) semantics. If  $\{\mathcal{ID}, \mathcal{CF2}\} \subseteq \mathbb{S}$ , Lemma 8 applies, in turn enabling the application of Lemma 2. This entails that  $\mathcal{ID}, \mathcal{CF2}, \mathcal{PR}, \mathcal{SST}$  are in agreement, thus  $\mathcal{AGR}(\mathbb{S}) \in \{\Sigma_4, \Sigma_5\}$ . If  $\{\mathcal{ID}, \mathcal{PR}\} \subseteq \mathbb{S} \wedge (\mathbb{S} \cap \{\mathcal{GR}, \mathcal{CO}, \mathcal{CF2}\}) = \emptyset$ , by Lemma 2 also  $\mathcal{SST}$  is in agreement, while  $\mathcal{ST}$  is in agreement if  $AF \in \mathcal{D}_{\mathcal{ST}}$ . This yields  $\mathcal{AGR}(\mathbb{S}) \in \{\Sigma_6, \Sigma_7\}$ . The last cases to be considered are included in the previous one. In fact, if  $\{\mathcal{ID}, \mathcal{SST}\} \subseteq \mathbb{S}$  then Lemma 9 applies, while if  $\{\mathcal{ID}, \mathcal{ST}\} \subseteq \mathbb{S}$ , by Proposition 1 also  $\mathcal{SST}$  is in agreement and we can apply again Lemma 9.

### 3.2. Multiple-status Agreement

The complete Venn diagram concerning all agreement classes is shown in Figure 2, where the bold rectangle represents the universe of all finite argumentation frameworks and again rectangular boxes represent classes of agreement and small ellipses represent single argumentation frameworks to be used as specific examples. As in the previous subsection, the diagram will be illustrated by examining first the set-theoretical relationships between the agreement classes depicted in Figure 2 and then proving that no other meaningful classes exist.

The first step will encompass the same three points as in the previous subsection. In particular, as to point 3, it can be noticed from Figure 2 that most intersections between classes correspond to unions of previously identified classes and/or differences between classes and can be easily determined by considering the sets of semantics involved. For this reason, only intersections requiring specific explanations (in particular all those concerning  $\Sigma_8$ ) will be explicitly discussed. Our analysis will now concern agreement classes involving only multiple-status semantics except  $\mathcal{CO}$ .

The smallest one includes all of them:  $\Sigma_9 \triangleq \mathcal{AGR}(\{\mathcal{PR}, \mathcal{CF2}, \mathcal{ST}, \mathcal{SST}\})$ . ( $\Sigma_9 \setminus \Sigma_4 \neq \emptyset$  as it includes  $AF_9 = \langle \{\alpha, \beta, \gamma, \delta\}, \{(\alpha, \beta), (\beta, \alpha), (\alpha, \gamma), (\beta, \gamma), (\gamma, \delta)\} \rangle$ ). Note

that  $\text{AF}_9 \in ((\Sigma_9 \setminus \Sigma_4) \cap \Sigma_8)$ . Also  $\Sigma_9 \setminus (\Sigma_4 \cup \Sigma_8)$  is not empty since it includes, for example,  $\text{AF}_{9'} = \langle \{\alpha, \beta, \gamma, \delta\}, \{(\alpha, \beta), (\beta, \alpha), (\alpha, \gamma), (\beta, \gamma), (\gamma, \delta), (\delta, \gamma)\} \rangle$ .

$\Sigma_{10} \triangleq \mathcal{AGR}(\{\mathcal{PR}, \mathcal{CF}2, \mathcal{SST}\})$  covers the case where stable extensions do not exist, while all other multiple-status semantics agree.  $\Sigma_{10} \setminus (\Sigma_9 \cup \Sigma_5) \neq \emptyset$  since it includes, for instance,  $\text{AF}_{10} = \langle \{\alpha, \beta, \gamma\}, \{(\alpha, \beta), (\beta, \alpha), (\gamma, \gamma)\} \rangle$ . Note that  $\text{AF}_{10} \in ((\Sigma_{10} \setminus (\Sigma_9 \cup \Sigma_5)) \cap \Sigma_8)$ . Also  $(\Sigma_{10} \setminus (\Sigma_9 \cup \Sigma_5 \cup \Sigma_8))$  is not empty since it includes, for example,  $\text{AF}_{10'} = \text{AF}_{10} \uplus \text{AF}_4$ .<sup>2</sup>

$\Sigma_{11} \triangleq \mathcal{AGR}(\{\mathcal{PR}, \mathcal{ST}, \mathcal{SST}\})$  coincides with the class of *coherent* argumentation frameworks considered in [3].  $(\Sigma_{11} \setminus (\Sigma_6 \cup \Sigma_9)) \neq \emptyset$  since it includes in particular  $\text{AF}_{11} = \langle \{\alpha, \beta, \gamma, \delta, \epsilon\}, \{(\alpha, \beta), (\alpha, \gamma), (\beta, \gamma), (\gamma, \alpha), (\delta, \epsilon), (\epsilon, \delta)\} \rangle$ . It can be noted that  $\text{AF}_{11} \notin \Sigma_8$ . Also  $(\Sigma_{11} \setminus (\Sigma_6 \cup \Sigma_9)) \cap \Sigma_8 \neq \emptyset$  since it includes  $\text{AF}_{11'} = \langle \{\alpha, \beta, \gamma, \delta\}, \{(\alpha, \beta), (\alpha, \gamma), (\alpha, \delta), (\beta, \gamma), (\beta, \delta), (\gamma, \alpha), (\delta, \alpha), (\delta, \beta), (\delta, \gamma)\} \rangle$ .

We are now left with three classes where only a pair of multiple-status semantics are in agreement. Let us start by considering  $\Sigma_{12} \triangleq \mathcal{AGR}(\mathcal{PR}, \mathcal{SST})$ .  $\Sigma_{12} \setminus (\Sigma_7 \cup \Sigma_{10} \cup \Sigma_{11}) \neq \emptyset$  as it includes  $\text{AF}_{12} = \langle \{\alpha, \beta, \gamma, \delta, \epsilon\}, \{(\alpha, \beta), (\beta, \alpha), (\gamma, \delta), (\delta, \epsilon), (\epsilon, \gamma)\} \rangle$ . It can be noted that  $\text{AF}_{12} \in \Sigma_8$ . For an example of argumentation framework included in  $\Sigma_{12} \setminus (\Sigma_7 \cup \Sigma_8 \cup \Sigma_{10} \cup \Sigma_{11})$  consider  $\text{AF}_{12'} = \text{AF}_{12} \uplus \text{AF}_4$ .

Finally  $\Sigma_{13} \triangleq \mathcal{AGR}(\mathcal{ST}, \mathcal{SST})$  coincides (by Proposition 1) with the class  $\mathcal{D}_{\mathcal{ST}}$  of argumentation frameworks where stable extensions exist, while the last pair to be considered corresponds to  $\Sigma_{14} \triangleq \mathcal{AGR}(\mathcal{PR}, \mathcal{CF}2)$ . The part of the diagram still to be illustrated involves argumentation frameworks outside  $\Sigma_{12}$  and requires an articulated treatment, since the intersections  $\Sigma_{13} \cap \Sigma_{14}$ ,  $\Sigma_{13} \cap \Sigma_8$ , and  $\Sigma_{14} \cap \Sigma_8$  do not allow a simple characterization in terms of the other identified classes. First the set difference  $\Sigma_{13} \setminus \Sigma_{12}$  can be partitioned into four non-empty subsets:

- $((\Sigma_{13} \setminus \Sigma_{12}) \cap \Sigma_{14} \cap \Sigma_8) \ni \text{AF}_{13} = \langle \{\alpha, \beta, \gamma\}, \{(\alpha, \beta), (\beta, \alpha), (\alpha, \gamma), (\gamma, \gamma)\} \rangle$ ;
- $((\Sigma_{13} \setminus \Sigma_{12}) \cap (\Sigma_8 \setminus \Sigma_{14})) \ni \text{AF}_{13'} = \text{AF}_{13} \uplus \text{AF}_{11'}$ ;
- $((\Sigma_{13} \setminus \Sigma_{12}) \cap (\Sigma_{14} \setminus \Sigma_8)) \ni \text{AF}_{13''} = \text{AF}_{13} \uplus \text{AF}_4$ ;
- $(\Sigma_{13} \setminus (\Sigma_{12} \cup \Sigma_{14} \cup \Sigma_8)) \ni \text{AF}_{13'''} = \text{AF}_{13} \uplus \text{AF}_4 \uplus \text{AF}_{11'}$ .

Then,  $\Sigma_{14} \setminus (\Sigma_{12} \cup \Sigma_{13})$  can be partitioned into two non-empty subsets:

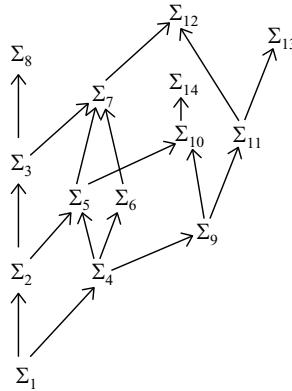
- $(\Sigma_{14} \setminus (\Sigma_{12} \cup \Sigma_{13} \cup \Sigma_8)) \ni \text{AF}_{14} = \text{AF}_{13} \uplus \text{AF}_4 \uplus \text{AF}_2$ ;
- $((\Sigma_{14} \setminus (\Sigma_{12} \cup \Sigma_{13})) \cap \Sigma_8) \ni \text{AF}_{14'} = \text{AF}_{13} \uplus \text{AF}_2$ .

Only the characterization of  $\Sigma_8$  remains to be completed, in fact  $\Sigma_8 \setminus (\Sigma_{12} \cup \Sigma_{13} \cup \Sigma_{14})$  is not empty since it includes  $\text{AF}_{8'} = \text{AF}_{13} \uplus \text{AF}_2 \uplus \text{AF}_{11'}$ . To conclude, argumentation frameworks where all semantics are in mutual disagreement also exist, like  $\text{AF}_{15} = \text{AF}_{13} \uplus \text{AF}_2 \uplus \text{AF}_4 \uplus \text{AF}_{11'}$ . The Hasse diagram corresponding to inclusion relationships between the agreement classes described above is shown in Figure 3 (where arrows point from subsets to supersets).

Now, to show that no other agreement classes involving  $\mathcal{CF}2$ ,  $\mathcal{ST}$ ,  $\mathcal{SST}$  and  $\mathcal{PR}$  are meaningful, the following Lemma is needed.

---

<sup>2</sup>While we have used the same labels  $\alpha, \beta, \dots$  to denote arguments of our sample argumentation frameworks, we implicitly assume that arguments with the same label in different argumentation frameworks are actually distinct. This assumption allows us to apply the combination operator  $\uplus$  to any pair of sample argumentation frameworks while keeping a simple notation. To make the distinction explicit, argument  $\alpha$  of  $\text{AF}_4$  should be actually labeled  $\alpha_4$ , argument  $\alpha$  of  $\text{AF}_{10}$  should be labeled  $\alpha_{10}$ , and so on, but we regard this as an unnecessary notational burden.



**Figure 3.** Inclusion relations between agreement classes.

**Lemma 10** If  $\mathcal{E}_{CF2}(\text{AF}) \subseteq \mathcal{AS}(\text{AF})$  then  $\mathcal{E}_{CF2}(\text{AF}) = \mathcal{EP}\mathcal{R}(\text{AF})$

*Proof.* To see that  $\mathcal{E}_{CF2}(\text{AF}) \subseteq \mathcal{EP}\mathcal{R}(\text{AF})$  we have to show that every  $CF2$  extension  $E$  is a maximal admissible set.  $E$  is admissible by the hypothesis. Moreover, since  $E \in \mathcal{MCF}_{\text{AF}}$  there can not be  $E' \supsetneq E$  which is conflict free and therefore there can not be  $E' \supseteq E$  which is admissible. To see that  $\mathcal{EP}\mathcal{R}(\text{AF}) \subseteq \mathcal{E}_{CF2}(\text{AF})$ , assume by contradiction that  $\exists E \in \mathcal{EP}\mathcal{R}(\text{AF}) \setminus \mathcal{E}_{CF2}(\text{AF})$ . It is known from [4] that  $\exists E' \in \mathcal{E}_{CF2}(\text{AF}) : E \subseteq E'$ . By the absurd hypothesis,  $E \subsetneq E'$ , but since  $E'$  is admissible this contradicts the fact that  $E$  is a maximal admissible set.  $\square$

Let us now consider all the possible relevant cases:

- $\{CF2, ST\} \subseteq \mathbb{S} \subseteq \{CF2, ST, PR, SST\} \Rightarrow \mathcal{AGR}(\mathbb{S}) = \Sigma_9$  (by Lemma 10 and Proposition 1);
- $\{CF2, SST\} \subseteq \mathbb{S} \subseteq \{CF2, PR, SST\} \Rightarrow \mathcal{AGR}(\mathbb{S}) = \Sigma_{10}$  (by Lemma 10);
- the only remaining case including  $CF2$  is  $\mathbb{S} = \{CF2, PR\}$  giving rise to  $\mathcal{AGR}(\mathbb{S}) = \Sigma_{14}$ ;
- if  $ST \in \mathbb{S}$  then by Proposition 1  $\mathcal{AGR}(\mathbb{S}) = \mathcal{AGR}(\mathbb{S} \cup \{SST\})$ : hence the only meaningful classes including  $ST$  and not including  $CF2$  are  $\Sigma_{11}$  and  $\Sigma_{13}$ ;
- if both  $ST$  and  $CF2$  are excluded the only remaining class is exactly  $\Sigma_{12} = \mathcal{AGR}(\{SST, PR\})$ .

#### 4. Conclusions

We have carried out a systematic analysis concerning classes of argumentation frameworks where semantics selected in a representative set of both traditional and more recent proposals agree. Fourteen meaningful classes have been identified out of 120 nominal ones and their set-theoretical relationships examined in detail, providing in particular an example of argumentation framework for each distinct region of the diagram in Figure 2. While the fact that only fourteen agreement classes exist can be regarded as a consequence (and confirmation) of the existence of common roots and shared basic intuitions underlying different proposals, it is also worth noting that there are argumentation frameworks like  $AF_{15}$  where all semantics considered in this paper are in mu-

tual disagreement. As to our knowledge, the issue of agreement between semantics has previously been considered in the literature only in relation to topological properties of argumentation frameworks. This kind of analysis has been first addressed in [3] where it is shown that a sufficient condition for agreement between grounded, preferred and stable semantics on an argumentation framework AF is that AF does not contain attack cycles, and a sufficient condition for agreement between preferred and stable semantics is that AF does not contain odd-length attack cycles. More recently, the special class of symmetric argumentation frameworks [9] (where every attack is mutual) has been shown to ensure agreement between preferred, stable and so-called naive semantics (actually coinciding with *CF2* semantics in this case). A more general analysis about topological classes of agreement has been carried out in [10] where, in particular, the notion of SCC-recursiveness [5] has been exploited. The analysis developed in the present paper is complementary to this research line since it considers classes of agreement derived only from properties of and relationships between the analyzed semantics, thus providing a systematic reference framework independent of any topological characterization. Analyzing relationships between these agreement classes and significant topological families of argumentation frameworks represents an interesting subject of current investigation.

## Acknowledgements

We are indebted to the anonymous referees for their helpful comments.

## References

- [1] M. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.
- [2] P. Baroni and M. Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10/15):675–700, 2007.
- [3] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
- [4] P. Baroni and M. Giacomin. Solving semantic problems with odd-length cycles in argumentation. In *Proc. 7th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2003)*, pages 440–451, Aalborg, Denmark, 2003. LNAI 2711, Springer-Verlag.
- [5] P. Baroni, M. Giacomin, and G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence*, 168(1-2):165–210, 2005.
- [6] P. Baroni and M. Giacomin. Evaluating argumentation semantics with respect to skepticism adequacy. In *Proc. 8th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, pages 329–340, Barcelona, E, 2005.
- [7] M. Caminada. Semi-stable semantics. In P. E. Dunne and T.J.M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Arguments (COMMA 2006)*, pages 121–130, Liverpool, UK, 2006. IOS Press.
- [8] P. M. Dung, P. Mancarella, and F. Toni. A dialectic procedure for sceptical, assumption-based argumentation. In P. E. Dunne and T.J.M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Arguments (COMMA 2006)*, pages 145–156, Liverpool, UK, 2006. IOS Press.
- [9] S. Coste-Marquis, C. Devred, and P. Marquis. Symmetric argumentation frameworks. In *Proc. 8th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, pages 317–328, Barcelona, E, 2005.
- [10] P. Baroni and M. Giacomin. Characterizing defeat graphs where argumentation semantics agree. In G. Simari and P. Torroni, editors, *Proc. of 1st Int. Workshop on Argumentation and Non-Monotonic Reasoning (ARGNMR07)*, pages 33–48, Tempe, AZ, 2007.

# Asking the Right Question: Forcing Commitment in Examination Dialogues

Trevor J. M. BENCH-CAPON<sup>a</sup>, Sylvie DOUTRE<sup>b</sup> Paul E. DUNNE<sup>a</sup>

<sup>a</sup> Department of Computer Science, The University of Liverpool, U.K.

<sup>b</sup> IRIT, University of Toulouse 1, France

**Abstract.** We introduce a new semantics for value-based argumentation frameworks (VAFs) – the uncontested semantics – whose principal motivation is as a mechanism with which to refine the nature of objective acceptance with respect to a given audience. The objectively accepted arguments of a VAF w.r.t. an audience  $\mathcal{R}$ , are those considered justified by all subscribing to the audience,  $\mathcal{R}$ , regardless of the specific value orderings that individuals may hold. In particular we examine how the concept of uncontested acceptance may be used in examination dialogues. The proposed semantics bear some aspects in common with the recently proposed *ideal semantics* for standard – i.e. value-free – argumentation frameworks. In this paper we consider applications of the new semantics to a specific “real” example and examine its relationship to the ideal semantics as well as analysing some basic complexity-theoretic issues.

**Keywords.** Computational properties of argumentation; Formalization of abstract argumentation; Dialogue based on argumentation; value-based argumentation; computational complexity

## Introduction

The notion of examination dialogues – dialogues designed not to discover what a person believes but rather their reasons for holding their beliefs - was introduced in [13]. Examples of such dialogues include traditional viva voce examinations, and political interviews. In both cases we want to discover more about the interviewee than simply what they will assent to. There is, however, the problem of choosing which question to ask: if the question is not well judged, there is the possibility of either evading the issue, or of offering a defence which makes no commitment to the underlying principles of the interviewee.

The value-based argumentation model introduced by Bench-Capon [4] has proven to be an important approach with which to examine such processes of practical reasoning and the rationale supporting why parties favour certain beliefs over alternatives, e.g. [15,5,1,2,14]. This model builds on the seminal approach to abstract argumentation pioneered by Dung [7] wherein a number of formalisations of the concept of “collection of justified beliefs” are proposed in terms of criteria defined on subsets of arguments in an abstract argumentation framework (AF). Two important classifications have been considered in terms of Dung’s basic

acceptability semantics: *credulous acceptance* – an argument is justified if it belongs to *at least one* admissible (i.e. self-defending and internally consistent) set of arguments; and *sceptical acceptance* – an argument is justified if it belongs to *every* maximal such set (or *preferred extension* in Dung’s terminology). In recent work Dung, Mancarella and Toni [8,9] advance a new classification, *ideal acceptance*, under which an argument has not only to be sceptically accepted but also contained within an admissible set of sceptically accepted arguments.

In the remainder of this article, we reprise the basic notions of Dung’s approach and Bench-Capon’s VAF development of this in Section 1 and formally introduce uncontested semantics. We then present a motivating example scenario in Section 2. The relationship between uncontested semantics and the ideal semantics of Dung *et al.* [8,9] is examined in Section 3. We give an overview of complexity-theoretic issues with uncontested semantics in Section 4 reporting results regarding both decision problems and the construction of uncontested sets in VAFs. For space reasons we eschew presentation of detailed proofs, full versions of all results are given in [11]. Section 5 considers the technical results with respect to examination dialogues. Further work and conclusions are discussed in Section 6.

## 1. Preliminaries: AFs and VAFs

The following concepts were introduced in Dung [7].

**Definition 1** An argumentation framework (AF) is a pair  $\mathcal{H} = \langle \mathcal{X}, \mathcal{A} \rangle$ , in which  $\mathcal{X}$  is a finite set of arguments and  $\mathcal{A} \subseteq \mathcal{X} \times \mathcal{X}$  is the attack relationship for  $\mathcal{H}$ . A pair  $\langle x, y \rangle \in \mathcal{A}$  is referred to as ‘ $y$  is attacked by  $x$ ’ or ‘ $x$  attacks  $y$ ’. The convention of excluding “self-attacking” arguments is assumed, i.e. for all  $x \in \mathcal{X}$ ,  $\langle x, x \rangle \notin \mathcal{A}$ . For  $R, S$  subsets of arguments in the AF  $\mathcal{H}(\mathcal{X}, \mathcal{A})$ , we say that  $s \in S$  is attacked by  $R$  – written  $\text{attacks}(R, s)$  – if there is some  $r \in R$  such that  $\langle r, s \rangle \in \mathcal{A}$ . For subsets  $R$  and  $S$  of  $\mathcal{X}$  we write  $\text{attacks}(R, S)$  if there is some  $s \in S$  for which  $\text{attacks}(R, s)$  holds;  $x \in \mathcal{X}$  is acceptable with respect to  $S$  if for every  $y \in \mathcal{X}$  that attacks  $x$  there is some  $z \in S$  that attacks  $y$ ;  $S$  is conflict-free if no argument in  $S$  is attacked by any other argument in  $S$ .

A conflict-free set  $S$  is admissible if every  $y \in S$  is acceptable w.r.t  $S$ ;  $S$  is a preferred extension if it is a maximal (with respect to  $\subseteq$ ) admissible set;  $S$  is a stable extension if  $S$  is conflict free and every  $y \notin S$  is attacked by  $S$ ;  $S$  is an ideal extension ([8,9]) of  $\mathcal{H}$  if  $S$  is admissible and a subset of every preferred extension of  $\mathcal{H}$ . We observe that [8,9] show that every AF has a unique maximal ideal extension (although, as with preferred extensions, this may be the empty set).

For  $S \subseteq \mathcal{X}$ ,

$$\begin{aligned} S^- &=_{\text{def}} \{ p : \exists q \in S \text{ such that } \langle p, q \rangle \in \mathcal{A} \} \\ S^+ &=_{\text{def}} \{ p : \exists q \in S \text{ such that } \langle q, p \rangle \in \mathcal{A} \} \end{aligned}$$

An argument  $x$  is credulously accepted if there is some preferred extension containing it;  $x$  is sceptically accepted if it is a member of every preferred extension.

The concepts of credulous and sceptical acceptance give rise to two decision problems –  $\text{CA}(\mathcal{H}, x)$  and  $\text{SA}(\mathcal{H}, x)$  – whose instances return positive answers whenever  $x$  is a member of at least one (resp. every) preferred extension of  $\mathcal{H}$ . The computational complexity of these has been studied in [6,12]. Bench-Capon [4] develops the concept of “attack” from Dung’s model to take account of *values*.

**Definition 2** A value-based argumentation framework (VAF), is defined by a triple  $\mathcal{H}^{(\mathcal{V})} = \langle \mathcal{H}(\mathcal{X}, \mathcal{A}), \mathcal{V}, \eta \rangle$ , where  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  is an AF,  $\mathcal{V} = \{v_1, v_2, \dots, v_k\}$  a set of  $k > 0$  values, and  $\eta : \mathcal{X} \rightarrow \mathcal{V}$  a mapping that associates a value  $\eta(x) \in \mathcal{V}$  with each argument  $x \in \mathcal{X}$ .

An audience for a VAF  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$ , is a binary relation  $\mathcal{R} \subset \mathcal{V} \times \mathcal{V}$  whose (irreflexive) transitive closure,  $\mathcal{R}^*$ , is asymmetric, i.e. at most one of  $\langle v, v' \rangle$ ,  $\langle v', v \rangle$  are members of  $\mathcal{R}^*$  for any distinct  $v, v' \in \mathcal{V}$ . We say that  $v_i$  is preferred to  $v_j$  in the audience  $\mathcal{R}$ , denoted  $v_i \succ_{\mathcal{R}} v_j$ , if  $\langle v_i, v_j \rangle \in \mathcal{R}^*$ . We say that  $\alpha$  is a specific audience if  $\alpha$  yields a total ordering of  $\mathcal{V}$ .

For an audience,  $\mathcal{R}$ , there will, generally, be a number of specific audiences consistent with  $\mathcal{R}^*$ . The notation,  $\chi(\mathcal{R})$  is used to describe the set of all such specific audiences, i.e.

$$\chi(\mathcal{R}) \quad =_{\text{def}} \quad \{ \alpha : \forall v, v' \in \mathcal{V} \langle v, v' \rangle \in \mathcal{R}^* \Rightarrow v \succ_{\alpha} v' \}$$

Following, [5, p. 40], the audience  $\mathcal{R} = \emptyset$  is called the universal audience by reason of  $\chi(\emptyset)$  containing every specific audience.

A standard assumption from [4] which we retain in our subsequent development is the *Multivalued Cycles Assumption* (MCA), i.e. “For any simple cycle of arguments in a VAF,  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$ , – i.e. a finite sequence of arguments  $y_1 y_2 \dots y_i y_{i+1} \dots y_r$  with  $y_1 = y_r$ ,  $|\{y_1, \dots, y_{r-1}\}| = r - 1$ , and  $\langle y_j, y_{j+1} \rangle \in \mathcal{A}$  for each  $1 \leq j < r$  – there are arguments  $y_i$  and  $y_j$  for which  $\eta(y_i) \neq \eta(y_j)$ .” In less formal terms, this assumption states every simple cycle in  $\mathcal{H}^{(\mathcal{V})}$  uses at least two distinct values.

Using VAFs, ideas analogous to those introduced in Defn. 1 are given by relativising the concept of “attack” using that of *successful* attack with respect to an audience. Thus,

**Definition 3** Let  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$  be a VAF and  $\mathcal{R}$  an audience. For arguments  $x, y$  in  $\mathcal{X}$ ,  $x$  is a successful attack on  $y$  (or  $x$  defeats  $y$ ) with respect to the audience  $\mathcal{R}$  if:  $\langle x, y \rangle \in \mathcal{A}$  and it is not the case that  $\eta(y) \succ_{\mathcal{R}} \eta(x)$ .

Replacing “attack” by “successful attack w.r.t. the audience  $\mathcal{R}$ ”, in Defn. 1 yields definitions of “conflict-free”, “admissible set” etc. relating to value-based systems, e.g.  $S$  is conflict-free w.r.t. to the audience  $\mathcal{R}$  if for each  $x, y$  in  $S$  it is not the case that  $x$  successfully attacks  $y$  w.r.t.  $\mathcal{R}$ . It may be noted that a conflict-free set in this sense is not necessarily a conflict-free set in the sense of Defn. 1: for  $x$  and  $y$  in  $S$  we may have  $\langle x, y \rangle \in \mathcal{A}$ , provided that  $\eta(y) \succ_{\mathcal{R}} \eta(x)$ , i.e. the value promoted by  $y$  is preferred to that promoted by  $x$  for the audience  $\mathcal{R}$ .

The concept of successful attack w.r.t. an audience  $\mathcal{R}$ , leads to the following notation as a parallel to the sets  $S^-$  and  $S^+$ . Given  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$ ,  $S \subseteq \mathcal{X}$  and an audience  $\mathcal{R} \subset \mathcal{V} \times \mathcal{V}$ ,

$$\begin{aligned} S_{\mathcal{R}}^- &= \text{def } \{ p : \exists q \in S \text{ such that } \langle p, q \rangle \in \mathcal{A} \text{ and } \langle \eta(q), \eta(p) \rangle \notin \mathcal{R}^* \} \\ S_{\mathcal{R}}^+ &= \text{def } \{ p : \exists q \in S \text{ such that } \langle q, p \rangle \in \mathcal{A} \text{ and } \langle \eta(p), \eta(q) \rangle \notin \mathcal{R}^* \} \end{aligned}$$

Bench-Capon [4] proves that every specific audience,  $\alpha$ , induces a unique preferred extension within its underlying VAF: for a given VAF,  $\mathcal{H}^{(\mathcal{V})}$ , we use  $P(\mathcal{H}^{(\mathcal{V})}, \alpha)$  to denote this extension: that  $P(\mathcal{H}^{(\mathcal{V})}, \alpha)$  is unique and can be constructed efficiently, is an easy consequence of the following fact, implicit in [4].

**Fact 1** *For any VAF,  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  (satisfying MCA) and specific audience  $\alpha$ , the framework induced by including only attacks in the set  $\mathcal{A}_\alpha$  given by  $\mathcal{A} \setminus \{\langle x, y \rangle : \eta(y) \succ_\alpha \eta(x)\}$  is acyclic.*

Analogous to the concepts of credulous and sceptical acceptance, in VAFs the ideas of *subjective* and *objective* acceptance (w.r.t. an audience  $\mathcal{R}$ ) arise, [5, p. 48].

**Subjective Acceptance:** (SBA)

**Instance:** A VAF,  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$ , argument  $x \in \mathcal{X}$ , audience  $\mathcal{R}$ .

**Question:** Is  $x \in P(\mathcal{H}^{(\mathcal{V})}, \alpha)$  for at least one specific audience  $\alpha \in \chi(\mathcal{R})$ ?

**Objective Acceptance:** (OBA)

**Instance:** A VAF,  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$ , argument  $x \in \mathcal{X}$ , audience  $\mathcal{R}$ .

**Question:** Is  $x \in P(\mathcal{H}^{(\mathcal{V})}, \alpha)$  for every specific audience  $\alpha \in \chi(\mathcal{R})$ ?

In this paper we are concerned with a VAF based extension semantics which we call the *uncontested semantics*.

**Definition 4** *Let  $\mathcal{H}^{(\mathcal{V})}$  be a VAF and  $\mathcal{R}$  an audience. A set of arguments,  $S$  in  $\mathcal{H}^{(\mathcal{V})}$  is an uncontested extension w.r.t.  $\mathcal{R}$  if it is an admissible set in  $\mathcal{H}(\langle \mathcal{X}, \mathcal{A} \rangle)$  and every argument in  $S$  is objectively acceptable in  $\mathcal{H}^{(\mathcal{V})}$  w.r.t. the audience  $\mathcal{R}$ .*

## 2. Motivating Example

We will use an example of political debate. In politics, support for a politician is typically not based solely on specific policies, but rather on the fundamental principles and values of the politicians, and the relative importance they give to them. In so far as we are in agreement with the values of the politicians, we can expect to be in agreement with their response to situations not yet encountered. We will base our treatment of the example on both AFS and VAFs. The latter allow us to distinguish between different audiences, characterised by the ordering they give to certain social values, and explore how different arguments will be acceptable to different audiences. Central to VAFs is the notion that arguments can be based on the promotion or demotion of social values: the manner in which this link can be made has been explored in the context of particular argument schemes in, e.g. [3].

It is assumed that the context supposes acceptance of the notion of a social contract, whereby citizens relinquish some of their freedom of action to the state, in return for assurances about protection of their lives and their property. We consider a debate which includes discussion of the death penalty for murder, and the inviolability of property rights.

Our argumentation framework will contain the following arguments:

- A1: *Human life is so important that the only appropriate penalty for murder is the death penalty.* This is the instinctive argument proposed by many supporters of capital punishment.
- A2: *Human life is so important that the State has no right to take it.* This is the kind of argument emphasising human rights that is advanced by organisation such as Amnesty International.

Both these arguments are based on the value that human life is of great importance, called hereafter  $L$ , for life. The following two arguments are based on the value of adherence to the Social Contract,  $C$ .

- A3: *Under the social contract the State must protect the lives of its citizens, and the death penalty is the only effective deterrent for murder.* This is the deterrence argument, often used to support capital punishment. In this context the deterrence is seen as an obligation required to honour the social contract.
- A4: *Under the social contract the State must protect the lives of its citizens, even wrongdoers, and so cannot take their lives itself.* This is also based on  $C$ . If it was argued that wrongdoers had breached the contract, the possibility of miscarriages of justice could be advanced, which means that inevitably some innocent people will suffer the death penalty.

The arguments  $\{A1, A2, A3, A4\}$  form a four-cycle:  $A2$  attacks  $A1$  which attacks  $A4$  which attacks  $A3$  which attacks  $A2$ .

Turning to property, we begin with the so called principle of necessity:

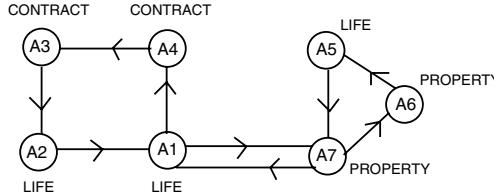
- A5: *A person may do whatever is necessary, including taking the property of another, to save their life.* Clearly this is based on the value  $L$ .

This can be opposed by a statement of the inviolability of property rights, based on the value of property rights,  $P$ , which in turned can be attacked by an argument based on the possibility of compensation, also based on  $P$ .

- A6: *No one has the right to take the property of another.*
- A7: *Any wrong done to a person can be cancelled by the wrongdoer paying the appropriate compensation.*

$A7$  is, however, attacked by  $A5$ , since it may be that the person concerned is unable to pay compensation. Thus  $A5$ ,  $A6$  and  $A7$  form a three cycle. We also have an attack relation between  $A1$  and  $A7$ : in some cultures a murderer may exculpate himself by paying a price (called an “eric” in ancient Ireland) to the victim’s family. The complete framework is shown in Fig 1.

Viewed as a Dung-style argumentation framework without reference to values, we have two preferred extensions:  $\{A1, A3, A6\}$  and  $\{A2, A4\}$ . Neither  $A5$  nor  $A7$  are even credulously accepted. Thus the maximal ideal extension is the empty set, since no argument in the framework is sceptically accepted. Thus the Dung-style framework in itself does not provide a useful basis for choosing between credulously accepted arguments. If, however, we consider the values, the



**Figure 1.** VAF relating to motivating example.

preferred extensions become  $\{A2, A4, A5, A7\}$  for audiences endorsing  $P \succ L$  and  $\{A2, A4, A5, A6\}$  for audiences containing  $L \succ P$ . Neither  $A1$  nor  $A3$  can be accepted no matter how the values are ranked. Thus  $\{A2, A4, A5\}$  are objectively acceptable, acceptable to all audiences, whereas  $A6$  and  $A7$  are subjectively acceptable, according to the preference between  $L$  and  $P$ .

Suppose now we wish to perform an examination dialogue on this framework: perhaps as part of a political interview. Our aim is to get the politician to reveal his ordering on values. It will not be appropriate to invite a defence of a subjectively acceptable argument, because such a question can be deflected as there is no obligation to defend the argument. Nor will it be fruitful to ask about the Dung-admissible arguments  $A2$  and  $A4$ , since these can be defended without making a commitment to one value over another. The only fruitful question is to ask for a defence of  $A5$ , since this must be accepted, and so must be defended, but defence requires either an appeal to  $A7$  to defeat  $A6$ , committing to  $P \succ L$  or to deny that  $A6$  defeats  $A5$ , committing to  $L \succ P$ .

A subset of objectively accepted arguments which define a Dung-admissible set – i.e. an uncontested extension as Defn 4 – is therefore of interest, since the arguments which are objectively accepted but which are not part of a Dung admissible set are precisely those arguments which may be fruitfully challenged in an examination dialogue.

### 3. Uncontested Semantics in VAFs

The uncontested extension can be seen as a value based analogue of the ideal extension [8,9]: the latter being defined as an admissible subset of *sceptically accepted* arguments; the former as an (Dung) admissible subset of those arguments *objectively accepted* w.r.t an audience  $\mathcal{R}$ .

Our main focus in this section is to review various properties of this approach: characteristics in common with ideal extensions such as that described in Thm. 1; as well as points under which these forms differ.

**Theorem 1** Let  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  be a VAF,  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  its supporting (value free) AF, and  $\mathcal{R} \subset \mathcal{V} \times \mathcal{V}$  be any audience. If  $\mathcal{U}_1$  and  $\mathcal{U}_2$  are both uncontested extensions of  $\mathcal{H}^{(\mathcal{V})}$  w.r.t.  $\mathcal{R}$  then  $\mathcal{U}_1 \cup \mathcal{U}_2$  is also an uncontested extension of  $\mathcal{H}^{(\mathcal{V})}$  w.r.t.  $\mathcal{R}$ .

**Corollary 1** For every VAF,  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  and audience  $\mathcal{R}$ , there is a unique, maximal uncontested extension w.r.t.  $\mathcal{R}$ .

We introduce the following notation for VAFs  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  and audiences  $\mathcal{R}$ .

- $\mathcal{U}_{\mathcal{R}} =$  The maximal uncontested extension of  $\mathcal{H}^{(\mathcal{V})}$  w.r.t.  $\mathcal{R}$
- $\mathcal{U} = \mathcal{U}_{\emptyset}$
- $\mathcal{M} =$  The maximal ideal extension of  $\mathcal{H}(\mathcal{X}, \mathcal{A})$

One issue arising from our definitions is to what extent do the ideal semantics and uncontested semantics give rise to *distinct* subsets. The results comprising this section consider this issue.

### Theorem 2

- a. *There are VAFs for which  $x \in \mathcal{M}$  but  $x \notin \mathcal{U}$ .*
- b. *There are VAFs for which  $x \in \mathcal{U}$  but  $x \notin \mathcal{M}$ .*
- c. *There are VAFs for which  $\emptyset \subset \mathcal{U} \subset \{x : \text{OBA}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle, x, \emptyset)\}$ .*

**Proof:** For (a), consider the AF formed by the three arguments  $\{A1, A7, A5\}$  from Fig. 1. The maximal ideal extension is  $\{A1, A5\}$  however  $\mathcal{U} = \{A5\}$  (since  $A1$  is not in the preferred extension with respect to the specific audience  $P \succ L$ ), so that  $A1 \in \mathcal{M}$  but  $A1 \notin \mathcal{U}$ . For (b), the VAF formed by the arguments  $\{A1, A2, A3, A4\}$  of Fig. 1 has  $\mathcal{U} = \{A2, A4\}$ : both specific audiences ( $L \succ C$  and  $C \succ L$ ) yielding the preferred extension  $\{A2, A4\}$ . In contrast  $\mathcal{M} = \emptyset$  in the underlying AF: both  $\{A2, A4\}$  and  $\{A1, A3\}$  being preferred extensions of this. Finally, for the VAF of Fig 1, we have  $\mathcal{U} = \{A2, A4\}$ , but the argument  $A5$  is also objectively accepted, so establishing (c).  $\square$

We have further indications that uncontested extensions describe radically different structures, in the failure of the following characterising lemmata, proven for ideal semantics in [10], to have an analogue in the uncontested semantics.

**Fact 2** Let  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  be an AF with maximal ideal extension  $\mathcal{M} \subseteq \mathcal{X}$ .

- a. *A subset  $S$  of  $\mathcal{X}$  defines an ideal extension of  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  if and only if both (I1) and (I2) below hold:*
  - I1.  *$S$  is an admissible set in  $\mathcal{H}(\mathcal{X}, \mathcal{A})$ .*
  - I2. *No attacker of  $S$  is credulously accepted, i.e.  $\forall y \in S^- \neg \text{CA}(\mathcal{H}, y)$ .*
- b. *For any  $x \in \mathcal{X}$ ,  $x \in \mathcal{M}$  if and only if both (M1) and (M2) below hold:*
  - M1. *No attacker of  $x$  is credulously accepted, i.e.  $\forall y \in \{x\}^- \neg \text{CA}(\mathcal{H}, y)$ .*
  - M2. *For each attacker,  $y$  of  $x$ , there is some attacker,  $z$  of  $y$ , for which  $z \in \mathcal{M}$ , i.e.  $\forall y \in \{x\}^- : \{y\}^- \cap \mathcal{M} \neq \emptyset$ .*

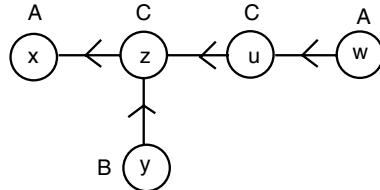
A natural reformulation of (M1) and (M2) in terms of VAFs is

- U1. No attacker of  $x$  is subjectively accepted w.r.t.  $\mathcal{R}$ .
- U2. For each attacker  $y$  of  $x$ , some attacker  $z$  of  $y$  is in  $\mathcal{U}_{\mathcal{R}}$ .

The following result demonstrates, however, that these fail to characterise maximal uncontested extensions.

**Lemma 1** There are VAFs,  $\mathcal{H}^{(V)}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  with maximal uncontested extensions,  $\mathcal{U}$ , that do not satisfy (U1). i.e. an argument  $y \in \mathcal{U}^-$  is subjectively accepted, so that this is not a necessary condition for membership in the maximal uncontested extension.

**Proof:** Consider the VAF,  $\mathcal{H}^{(V)}$  of Fig. 2



**Figure 2.** No attacker subjectively accepted is not a necessary condition

The maximal uncontested extension is formed by the set  $\{x, y, w\}$ : this is easily seen to be admissible since  $\{y, w\}^- = \emptyset$  and the sole attacker,  $z$ , of  $x$  is counterattacked by  $y$ . Each argument in  $\{x, y, w\}$  is also objectively accepted: that  $\{y, w\} \subseteq P(\mathcal{H}^{(V)}, \alpha)$  for any specific audience  $\alpha$ , is immediate from  $\{y, w\}^- = \emptyset$ . The argument  $x$  is in  $P(\mathcal{H}^{(V)}, \alpha)$  for all specific audiences in which  $A \succ_\alpha C$  (since the attack  $\langle z, x \rangle$  does not succeed); the remaining specific audiences (in which  $C \succ_\alpha A$ ) satisfy  $u \in P(\mathcal{H}^{(V)}, \alpha)$ , so that  $\langle u, z \rangle$  is an attack in the acyclic AF induced by these, thereby providing  $u$  as a defence to the attack by  $z$  on  $x$ .

The argument  $z \in \{x, y, w\}^-$  is, however, subjectively accepted using the specific audience  $A \succ C \succ B$ :  $\langle y, z \rangle$  does not succeed with respect to this audience; the (successful) attack  $\langle w, u \rangle$  provides  $w$  as a defence to the attack by  $u$  on  $z$  so that  $P(\mathcal{H}^{(V)}, A \succ C \succ B) = \{x, y, w, z\}$ .  $\square$

We observe that the constructions of Thm. 2 and Lemma 1 further emphasise the fact that there are a number of subtle differences between the divers acceptability semantics proposed for VAFS – i.e. Subjective, Objective and Uncontested – in comparison with superficially similar acceptability semantics in AFS, i.e. Credulous, Sceptical and Ideal. Thm. 2 and Lemma 1 thus develop the related comparison of Bench-Capon *et al.* [5, Thm. 12, pp. 50–51].

Despite Fact 2 failing to have an immediate counterpart when characterising uncontested extensions, cf. Lemma 1, it turns out that a very similar result can be obtained, albeit in a rather indirect manner. We first introduce the notion of a VAF being *k*-terse, where  $k \geq 1$ .

**Definition 5** For  $k \in \mathbb{N}$ , the VAF,  $\mathcal{H}^{(\mathcal{V})}((\mathcal{X}, \mathcal{A}, \mathcal{V}, \eta))$  is  $k$ -terse if every simple directed path of length  $k$  involves at most  $k$  different values from  $\mathcal{V}$ . Formally,  $\forall x_1 x_2 x_3 \cdots x_{k+1} \in \mathcal{X}^{k+1}$  such that  $\langle x_i, x_{i+1} \rangle \in \mathcal{A}$  for each  $1 \leq i \leq k$  and  $|\{x_1, x_2, \dots, x_{k+1}\}| = k+1$ ,  $|\{\eta(x_i) : 1 \leq i \leq k+1\}| \leq k$ .

**Theorem 3** Let  $\mathcal{H}^{(\mathcal{V})}(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle)$  be any 2-terse VAF,  $\mathcal{R}$  an audience. The argument  $x \in \mathcal{X}$  is in  $\mathcal{U}_{\mathcal{R}}$  if and only if both of the following hold:

- U1. No attacker,  $y$ , of  $x$  is subjectively accepted w.r.t.  $\mathcal{R}$  in  $\mathcal{H}^{(\mathcal{V})}$ , i.e.  $\forall y \in \{x\}^-, \neg \text{SBA}(\mathcal{H}^{(\mathcal{V})}, y, \mathcal{R})$ .
- U2. For every attacker,  $y$ , of  $x$ , at least one attacker,  $z$  of  $y$ , is in  $\mathcal{U}_{\mathcal{R}}$ , i.e.  $\forall y \in \{x\}^- \setminus \{y\}^- \cap \mathcal{U}_{\mathcal{R}} \neq \emptyset$ .

The property 2-terseness may seem rather too restrictive in order for the characterisation of Thm. 3 to be widely applicable. As the following result (whose proof is given in [11]) shows, this in fact is not necessarily the case.

**Lemma 2** (Path Dilution Lemma – PDL) *Let  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$  be any VAF. There is a VAF,  $\langle \mathcal{X} \cup \mathcal{Y}, \mathcal{B}, \mathcal{V}, \varepsilon \rangle$  such that*

- PD1.  $\forall x \in \mathcal{X}, \forall \alpha, x \in P(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle, \alpha) \Leftrightarrow x \in P(\langle \mathcal{X} \cup \mathcal{Y}, \mathcal{B}, \mathcal{V}, \varepsilon \rangle, \alpha)$ .
- PD2.  $\langle \mathcal{X} \cup \mathcal{Y}, \mathcal{B}, \mathcal{V}, \varepsilon \rangle$  is 2-terse.
- PD3.  $\langle \mathcal{X} \cup \mathcal{Y}, \mathcal{B}, \mathcal{V}, \varepsilon \rangle$  satisfies MCA if and only if  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$  does so.

Furthermore  $\langle \mathcal{X} \cup \mathcal{Y}, \mathcal{B}, \mathcal{V}, \varepsilon \rangle$  is constructible in polynomial time from  $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$ .

#### 4. Complexity of Uncontested Semantics

Results on the computational complexity of problems in the *ideal semantics* of Dung *et al.* [8,9] are presented by Dunne in [10]. These decision problems and their counterparts in uncontested semantics are described in Table 1.

**Table 1.** Decision questions for Ideal and Uncontested Semantics

Problem Name	Instance	Question
IE	$\langle \mathcal{H}(\mathcal{X}, \mathcal{A}), S \rangle$	Is $S$ an ideal extension?
IA	$\langle \mathcal{H}(\mathcal{X}, \mathcal{A}), x \rangle$	Is $x$ in the maximal ideal extension?
MIE $_{\emptyset}$	$\mathcal{H}(\mathcal{X}, \mathcal{A})$	Is the maximal ideal extension empty?
MIE	$\langle \mathcal{H}(\mathcal{X}, \mathcal{A}), S \rangle$	Is $S$ the <i>maximal</i> ideal extension?
UE	$\langle \mathcal{H}^{(\mathcal{V})}, S, \mathcal{R} \rangle$	Is $S$ an uncontested extension?
UA	$\langle \mathcal{H}^{(\mathcal{V})}, x, \mathcal{R} \rangle$	Is $x \in \mathcal{U}_{\mathcal{R}}(\mathcal{H}^{(\mathcal{V})})$ ?
MUE $_{\emptyset}$	$\langle \mathcal{H}^{(\mathcal{V})}, \mathcal{R} \rangle$	Is $\mathcal{U}_{\mathcal{R}}(\mathcal{H}^{(\mathcal{V})}) = \emptyset$ ?
MUE	$\langle \mathcal{H}^{(\mathcal{V})}, S, \mathcal{R} \rangle$	Is $S = \mathcal{U}_{\mathcal{R}}(\mathcal{H}^{(\mathcal{V})})$ ?

The complexity class  $\text{FP}_{\parallel}^{\text{NP}}$  comprises those *function* computations realised by deterministic polynomial time algorithms that make a polynomially bounded number of *queries* to an NP oracle with all queries performed in parallel, i.e. non-adaptively so that the form of each query must be determined in advance of any invocation of the NP oracle.

#### Theorem 4

- UE is co-NP-complete even if  $\langle \mathcal{X}, \mathcal{A} \rangle$  is a binary tree.
- UA is co-NP-hard even if  $\langle \mathcal{X}, \mathcal{A} \rangle$  is a binary tree.
- MUE $_{\emptyset}$  is NP-hard.
- MUE is D<sup>P</sup>-hard.

- e. Let FMUE be the (single-valued) function, defined as

$$\text{FMUE}(\mathcal{H}^{(\mathcal{V})}((\mathcal{X}, \mathcal{A}, \mathcal{V}, \eta)), \mathcal{R}) =_{\text{def}} \mathcal{U}_{\mathcal{R}}(\mathcal{H}^{(\mathcal{V})})$$

i.e. given a VAF,  $\mathcal{H}^{(\mathcal{V})}$ , and audience,  $\mathcal{R}$ , the function FMUE returns the maximal uncontested extension w.r.t  $\mathcal{R}$ : FMUE is  $\text{FP}_{||}^{\text{NP}}$ -complete.

- f. UA,  $\text{MUE}_{\emptyset}$  and MUE are all in  $\text{P}_{||}^{\text{NP}}$ .

- g. If UA is NP-hard then UA is  $\text{P}_{||}^{\text{NP}}$ -complete.

## 5. Discussion

In this section we will try to relate these theoretical results to our original motivation, which was to find the most appropriate question to ask in the context of an examination dialogue intended to elicit the value preferences of the person being questioned. For this, ideally, we need to find an argument which is objectively accepted, but not part of the uncontested extension. The results obtained in the preceding section suggest that this is not in general an easy problem: given a VAF, we do not have an efficient algorithm guaranteed to deliver the arguments we want. That the problem is difficult is, perhaps, unsurprising: the inconclusive nature of the vast majority of interviews with politicians that we have all witnessed leads one us to suspect that killer questions are often not obvious. Heuristics could be used and we examine some in the context of political debate.

First it can be reasonable to assume that a subset of  $\mathcal{X}_{\text{OBA}}$  (the set of objectively accepted arguments) is known at the outset: there are some theses that are part of the received wisdom, or the political consensus, hence reasonable to expect the politician to be committed to defending. Note that the audience to which they are acceptable may not be the universal audience, but perhaps only that of the grouping to which the politician subscribes, e.g. when we try to place the politician within a particular faction. In the latter case manifesto commitments, subscribed to by the party as a whole, perhaps for different reasons, would provide a good starting choice of  $S \subseteq \mathcal{X}_{\text{OBA}}$ . So, can we select one such argument to use in our examination dialogue? It is easy to test their admissibility and, if they fail to be so, to identify argument(s) that are unacceptable w.r.t. the set. If the set,  $S$  (known to be a subset of  $\mathcal{X}_{\text{OBA}}$ ) is, itself, admissible then its members are unsuitable for our purposes. It does not, however, follow that if they do not form an admissible set, that the arguments preventing admissibility are appropriate. For there may be other objectively acceptable arguments which could be added to make the set admissible. Finding these, however, is still intractable.

One approach is to identify common properties of objectively accepted arguments that are outside the uncontested extension. Odd-length cycles provide one fruitful source, but such arguments need *not* form part of an odd cycle. There is a simple counterexample refuting this: a chain of four arguments –  $x \rightarrow y \rightarrow z \rightarrow u$  with  $\eta(x) = \eta(u) = A$  and  $\eta(y) = \eta(z) = B$ , so that  $\mathcal{X}_{\text{OBA}} = \{x, u\}$ ,  $\mathcal{U} = \{x\}$  and  $u$  is not contained in any cycle. Note, however, that the *dialectical* structure is, as with the odd cycle, values  $A$  attacked by  $B$  attacked by  $B$  attacked by  $A$ .

An alternative is to set up the dialogue by preparing the framework so that it does contain the desired kind of argument. For example suppose we have the

following argument which forms part of the political consensus, or at least seems acceptable to all UK political parties:

*PA1* We should not raise taxes, as people should be able to spend their money as they please, which promotes the value of choice.

The argument *PA1* attacks the following argument:

*PA2* We should increase spending on public services to make top quality health care available to all to promote social justice. This requires a rise in taxes.

This argument could be accepted together with *PA1*, since we could argue that social justice was so important that despite the reduction of choice we should raise taxes to pay for better health care. If asked to defend *PA2* in the light of the consensus on *PA1*, however, a skilful politician will have no difficulty in deflecting the question by talking of how a responsible government must not shy away from hard choices and how it is all a question of priorities.

Suppose, however, that we introduce a third argument:

*PA3* Social justice demands that the ability to make choices should not depend upon income, and so income must be redistributed through higher taxes.

This attacks *PA1*, since it requires redistribution. It is, however, attacked by *PA2*, since there the spending demanded is hypothesized to a particular purpose, so that the increased taxes are not used to effect redistribution. Effectively *PA2* urges the reduction choice for all, while not affecting its unequal distribution.

*PA1* remains objectively acceptable, since it is the argument with a distinct value in a three cycle. But now, since it is under attack, it needs to be defended and this will require either preferring choice, so that it is regrettable but unavoidable that choices remain unevenly distributed, or preferring social justice, so that equity in health care is achieved. Although this means that taxes must rise, they are to be used for other purposes than redistributing the degree of choice. Thus the politician must choose between a paternalistic, interventionist approach, in which decisions are made for people (albeit in what is conceived of their own best interests), and a laissez-faire approach in which those who have the requisite means are allowed to make their own choices.

What this in turn means is that someone conducting an examination dialogue should not necessarily expect to have the materials needed readily available at the outset: and even if they are there, it may be impossible to identify them. It may, therefore, be better to use some degree of creativity in order to provide the arguments necessary to make the interrogation effective. What the notion of an uncontested extension does allow us to see is the kind of structures we need to create in order to force the distinctions we wish to explore to emerge.

## 6. Conclusions and Further Development

This article presents an extension-based semantics for value-based argumentation frameworks which arises as a natural counterpart to the ideal semantics of Dung *et al.*

al. [8,9] for the standard argumentation frameworks of [7]. It has been shown that although, in common with the ideal semantics, this form satisfies the property of defining a unique maximal extension w.r.t. any audience  $\mathcal{R}$ , nevertheless these give rise to significantly different behaviours: in general, these semantics are not coincident for a given VAF even in the case of the universal audience  $\mathcal{R} = \emptyset$ .

The motivation underlying our proposed formulation is in order to present one mechanism by which the nature of *objective acceptability* in VAFs may be further refined in the sense that those objectively accepted arguments falling outside the unique maximal uncontested extensions, although accepted by all relevant audiences, are so accepted on account of differing reasoning patterns germane to the audiences concerned. Such arguments are of interest as being the most fruitful starting points for examination dialogues.

## References

- [1] K. Atkinson. Value-based argumentation for democratic decision support. In P. E. Dunne and T. J. M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Argument*, volume 144 of *FAIA*, pages 47–58. IOS Press, 2006.
- [2] K. Atkinson and T. J. M. Bench-Capon. Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artificial Intelligence*, 171:855–874, 2007.
- [3] K. Atkinson, T. J. M. Bench-Capon, and P. McBurney. Computational representation of practical argument. *Synthese*, 152:157–206, 2006.
- [4] T. J. M. Bench-Capon. Persuasion in Practical Argument Using Value-based Argumentation Frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
- [5] T. J. M. Bench-Capon, S. Doutre, and P. E. Dunne. Audiences in argumentation frameworks. *Artificial Intelligence*, 171:42–71, 2007.
- [6] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science*, 170:209–244, 1996.
- [7] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, 77:321–357, 1995.
- [8] P. M. Dung, P. Mancarella, and F. Toni. A dialectical procedure for sceptical assumption-based argumentation. In P. E. Dunne and T. J. M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Argument*, volume 144 of *FAIA*, pages 145–156. IOS Press, 2006.
- [9] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171:642–674, 2007.
- [10] P. E. Dunne. The computational complexity of ideal semantics I: abstract argumentation frameworks. Technical Report ULCS-07-008, Dept. of Comp. Sci., Univ. of Liverpool, June 2007.
- [11] P. E. Dunne. Uncontested semantics for value-based argumentation. Technical Report ULCS-07-013, Dept. of Comp. Sci., Univ. of Liverpool, July 2007.
- [12] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141:187–203, 2002.
- [13] P. E. Dunne, S. Doutre, and T. J. M. Bench-Capon. Discovering inconsistency through examination dialogues. In *Proc. IJCAI05*, pages 1560–1561, 2005.
- [14] S. Kaci and L. van der Torre. Preference based argumentation: arguments supporting multiple values. *Int. Jnl. of Approximate Reasoning*, 2007. (in press).
- [15] S. Modgil. Value based argumentation in hierarchical argumentation frameworks. In P. E. Dunne and T. J. M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Argument*, volume 144 of *FAIA*, pages 297–308. IOS Press, 2006.

# Ontological Foundations for Scholarly Debate Mapping Technology

Neil BENN\*, Simon BUCKINGHAM SHUM, John DOMINGUE, Clara MANCINI  
*Knowledge Media Institute, Centre for Research in Computing, The Open University*

**Abstract.** Mapping scholarly debates is an important genre of what can be called Knowledge Domain Analytics (KDA) technology – i.e. technology which combines both quantitative and qualitative methods of analysing specialist knowledge domains. However, current KDA technology research has emerged from diverse traditions and thus lacks a common conceptual foundation. This paper reports on the design of a KDA ontology that aims to provide this foundation. The paper then describes the argumentation extensions to the ontology for supporting scholarly debate mapping as a special form of KDA and demonstrates its expressive capabilities using a case study debate.

**Keywords.** Scholarly Debate Mapping, Macro-argument Analysis, Ontologies

## Introduction

Research into tools to support both quantitative and qualitative analysis of specialist knowledge domains has been undertaken within the two broadly independent traditions of *Bibliometrics* and *Knowledge Management*. Knowledge Domain Analysis (KDA) tools within the first tradition (e.g. CiteSeer [1] and CiteSpace [2]) follow a citation-based approach of representing knowledge domains, where citation links are used as the basis for identifying structural patterns in the relationships among authors and publications. Tools within the second tradition (e.g. Bibster [3], ESKIMO [4], CS AKTIVE SPACE [5], and ClaiMaker [6]) extend the representational approach to include more features of knowledge domains – e.g. the types of agents or actors in the domain, their affiliations, and their research activities – with the aim of enabling more precise questions to be asked of the domain. This second approach depends on the development of software artefacts called ontologies, which are used to explicitly define schemes for representing knowledge domains.

This paper describes exploratory research into how these two traditions can be bridged in order to exploit both the benefit of ontologies to enable more feature-rich representations, as well as the established techniques of Bibliometrics for identifying structural patterns in the domain. The first section describes the design of a merged KDA ontology that integrates the existing ontologies specified in [3]- [6] (§1). Next, the paper describes how the merged ontology can be extended to include both a scheme for representing scholarly debates and inference rules for reasoning about the debate (§2). Thirdly, the extended ontology is applied to the representation and analysis of the

---

\* Correspondence to: Neil Benn, Knowledge Media Institute, The Open University, MK7 6AA, UK. Tel: +44 (0) 1908 695837; Fax: +44 (0) 1908 653196; E-mail: [n.j.l.benn@open.ac.uk](mailto:n.j.l.benn@open.ac.uk)

abortion debate, which demonstrates the benefits of reusing techniques from both traditions (§3). The key lessons from this research are then discussed (§4), before concluding with directions for future research (§5).

## 1. A merged KDA ontology

One method for merging heterogeneous ontologies requires that the existing ontologies are aligned to a more generic reference ontology that can be used to compare the individual classes in the existing ontologies. Based on the fundamental assumptions that knowledge representation and communication are major activities of knowledge domains, and that knowledge representation and communication constitute semiotic activities, we propose to align the existing ontologies to a reference ontology that describes the interactions between components of any given semiotic activity.

### 1.1. Invoking a generic theory of semiotics as a reference framework

Semiotics is the study of signs and their use in representation and communication. According to Peirce's theory of semiotics [7], the basic sign-structure in any instance of representation and communication consists of three components: (1) the *sign-vehicle*, (2) the *object* referred to by the sign-vehicle, and (3) the *interpretant*, which is the mental representation that links the sign-vehicle to the object in the mind of some conceiving agent.

Recent research within the ontology engineering field has introduced a reusable *Semiotic Ontology Design Pattern (SemODP)* [8] that specifies, with some variation of Peircean terminology, the interactions between components of any given semiotic activity. The SemODP<sup>1</sup> is shown in Figure 1.

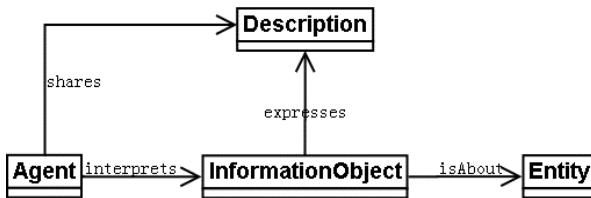


Figure 1. The Semiotic Ontology Design Pattern

In Peircean terminology, the *InformationObject* class of the SemODP represents the 'sign-vehicle'. In the context of knowledge domains, the most typical examples of information objects are publications, which are the main vehicles of knowledge representation and communication. A single publication can be regarded as an Information Object, as can each clause, sentence, table, graph, and figure that is either a verbal or non-verbal expression of knowledge within a publication.

The SemODP classes *Description* and *Entity* respectively correspond to the 'interpretant' and the 'object' in Peircean terminology. The *Description* class is the abstract, communicable knowledge content that an information object expresses. For

<sup>1</sup> The SemODP pattern shown here is based on an updated version of the 'home' ontology from which it is extracted (accessible at <http://www.loa-cnr.it/ontologies/cDnS.owl>)

example, a single publication is an information object that expresses a thesis (in much the same manner as a novel expresses a particular plot). The Entity class covers any physical or non-physical entity that an Information Object refers to via the ‘is about’ relation.

Finally, the SemODP specifies the *Agent* class. An agent is required to interpret a given Information Object and in such a case, the agent is said to conceive the Description expressed by that particular Information Object. In knowledge domains, instances of the Agent class include both “agentive physical objects” such as persons and “agentive social objects” such as organisations.

### 1.2. The core KDA ontology classes

The SemODP is used to merge the existing ontologies through a process of aligning the classes in the existing ontologies to the SemODP classes. The process also reveals the core KDA ontology classes that are based on consensus across the existing ontologies. For example, the consensus classes across the existing ontologies that can play the role of ‘Agent’ are ‘Person’ and ‘Organisation’. However, it should be noted that core classes are not fixed indefinitely and constantly evolve as applications generate experience and consensus changes about what is central [9]. Table 1 shows the core KDA classes and their relationships to SemODP classes as well as existing ontology classes.

**Table 1.** The SemODP classes, the existing KDA classes they subsume, and the core KDA classes with their main properties.

SemODP class	Existing ontology classes <sup>2</sup>	Core KDA class	Properties (Type)
cdns:Agent	swrc:Person, eskimo:Person, akt:Person	mkda:Person	name (String)
cdns:Agent	swrc:Organisation, eskimo:Organisation, akt:Organisation	mkda:Organisation	name (String)
cdns:InformationObject	swrc:Publication, eskimo:Publication, akt:Publication	mkda:Publication	has Author (Agent); hasTitle (String); expresses (Description)
cdns:Description	scholonto:Concept	mkda:PropositionalContent	verbalExpression (String)
cdns:Description	scholonto:Concept	mkda:NonPropositionalContent	verbalExpression (String)

<sup>2</sup>‘swrc:’ is the prefix for the ontology of the Bibster tool [3]; ‘eskimo:’ is the prefix for the ontology of the ESKIMO tool [4]; ‘akt:’ is the prefix for the ontology of the CS AKTIVE SPACE tool [5]; ‘scholonto:’ is the prefix for the ontology of the ClaiMaker tool [6].

## 2. Extending the ontology to support scholarly debate mapping

The work in this section builds on the debate mapping approach of Robert Horn [10] who has produced a classic series of seven debate maps for analysing the history and current status of debate on whether computers can think<sup>3</sup>. What has emerged from this debate mapping approach is a theory of the structure of debate, which has subsequently been articulated by Yoshimi [11] in what he calls a “logic of debate”. Whereas most argumentation research concentrates on the *microstructure* of arguments (e.g. the types of inference schemes for inferring conclusions from premises), the concern of a logic of debate is how arguments themselves are “constituents in *macro*-level dialectical structures” [11]. The basic elements of this logic of debate are implemented as additional classes and relations in the merged KDA ontology.

### 2.1. Debate representation

#### *Issues*

In the proposed logic of debate, *issues* can be characterised as the organisational atoms in structuring scholarly debate<sup>4</sup>. Indeed, according to [13], one of the essential characteristics of argumentation is that there is an issue to be settled and that the argumentative reasoning is being used to contribute to a settling of the issue. An *Issue* class is introduced into the ontology as a specialisation of the core KDA class *NonPropositionalContent*.

#### *Propositions & Arguments*

The other basic elements in the proposed logic of debate are claims (propositions) and arguments, where the term ‘argument’ is used in the abstract sense of a set of propositions, one of which is a conclusion and the rest of which are premises. However, an argument can play the role of premise in another argument, thus allowing the chaining of arguments. *Proposition* and *Argument* classes are introduced into the ontology as specialisations of the core KDA class *PropositionalContent*. The main relations between claims and arguments in the logic of debate are *supports* and *disputes*.

### 2.2. Debate analysis

Analysing a debate involves reasoning with representations of that debate to detect potentially significant features of the debate. Here we propose to draw on the well-developed network-based analytical techniques employed within the Bibliometrics tradition. However, this reuse is not straightforward since the analytical techniques are typically designed to operate on single-link-type network representations of domains, where the links between nodes are used to signal *positive association* between nodes. For example, network-based analytical techniques are often applied to co-citation networks where a link between two publications is established when they are both cited by a common third publication, and that link signals positive association between the two publications.

This single-link-type assumption presents a challenge because the ontology-based debate representations can be regarded as ‘multi-faceted’ representations – i.e. there are

---

<sup>3</sup> This debate largely takes place within the knowledge domain of *Artificial Intelligence*

<sup>4</sup> This concurs with Rittel’s argument [12] that issues serve as the “organisational atoms” of *Issue-Based Information Systems* (IBIS) for tackling “wicked” sociotechnical design problems.

a number of node types and a number of link types. Thus, before network-based analytical techniques can be reused for debate analysis, transformation rules need to be defined that can project the ‘multi-faceted’ debate representation onto a single-link-type representation.

In order for such a projection to work, the multiple relations in the ontology need to be uniformly interpreted from a single perspective. We propose to interpret the relations in the ontology in a ‘rhetorical-discourse’ context. Indeed, it can be argued that the analytical techniques of the Bibliometric tradition interpret the publications in a citation-based network as rhetorical viewpoints, which implies that the positive association link between nodes can be interpreted as rhetorical agreement.

Furthermore, the work of Mancini and Buckingham Shum [14] provides the basis for an efficient implementation of the transformation rules. An important feature of their work is the use of a limited set of cognitively grounded parameters (derived from the psycholinguistic work on discourse comprehension<sup>5</sup> by [15]) to define the underlying meaning of discourse relations in the ontology of the ClaiMaker tool. Mancini and Buckingham Shum [14] anticipate that using discourse coherence parameters as the underlying definition language will allow different discourse-relation vocabularies to be used for representing discourse without changing the underlying discourse analysis services provided by their tool. The four bipolar discourse parameters proposed by [15] are: *Additive/Causal*, *Positive/Negative*, *Semantic/Pragmatic*, and *Basic/Non-Basic*.

The ‘Additive/Causal’ parameter depends, respectively, on whether a weak or strong correlation exists between two discourse units. Note that ‘causal’ is generally given a broad reading in discourse comprehension research to include causality involved in argumentation (where a conclusion is motivated *because* of a particular line of reasoning), as well as more typical cause-effect relationships between states of affairs.

The ‘Positive/Negative’ parameter depends, respectively, on whether or not the *expected* connection holds between the two discourse units in question. For example, in the sentence “*Because he had political experience, he was elected president*” the connection between the two units is Positive since the reader would typically expect “being elected president” to follow from “having political experience”. However, in the sentence “*He did not have any political experience, yet he was elected president*” the connection between the two units is Negative since the expected consequent of “not having any political experience” is “*not* being elected president”, but what is actually expressed is a violation of that expectation – i.e. “*yet he was elected president*”.

The ‘Semantic/Pragmatic’ parameter depends, respectively, on whether the connection between the two discourse units lies between their factual content or between the speech acts of expressing the two discourse units. At this stage we are primarily focussed on enabling debate analysis, and since debate analysis falls within the realm of speech acts [16], the relations between entities that make up a debate representation will be parameterised as ‘Pragmatic’ by default.

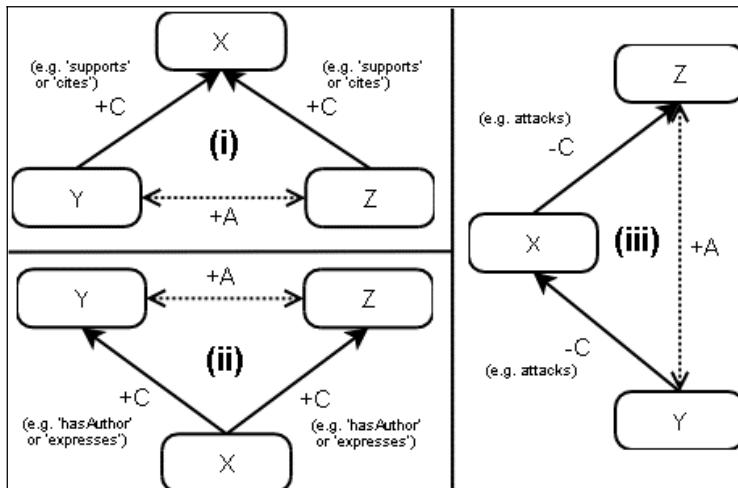
The ‘Basic/Non-Basic’ parameter depends, respectively, on whether or not, in the case of Causal connections, the cause precedes the consequent in the presentation of the discourse. The example – “*Because he had political experience, he was elected president*” – is parameterised as Basic, whereas the example – “*He was elected*

---

<sup>5</sup> Discourse comprehension research in general is concerned with the process by which readers are able to construct a *coherent* mental representation of the information conveyed by a particular piece of discourse.

*president because he had political experience*" – is parameterised as Non-Basic. This parameter is largely about presentation and does not affect the essential nature or meaning of the discourse connection. Thus it can be omitted from the basic parameterisation of relations in the ontology.

These coherence parameters are then used as a grammar for defining relations in the merged KDA ontology, including relations between publications, between persons, and between arguments. The benefit of this approach is that rather than implement a multitude of inference rules for inferring positive association, only a limited set of parameterised inference rules need to be implemented. For example, Figure 2(i), which shows a parameterised rule for inferring a +ADDITIVE connection between some  $Y$  and some  $Z$ , covers typical positive association inferences such as when two arguments support a common third argument or when two publications cite a common third publication. Figure 2(ii), which also shows a parameterised rule, covers typical positive association inferences such as when two persons author a common publication or when two arguments are expressed by a common publication. Finally, Figure 2(iii) covers a typical 'undercutting' pattern in argument analysis which is a variation of the social network analysis adage that "the enemy of my enemy is my friend".



**Figure 2.** Some of the CCR-parameterised inference rules in the ontology. The dotted line indicates that a +ADDITIVE connection is inferred based on the other connections.

### 3. The abortion debate case study

This section briefly describes the representation and analysis of the debate about the desired legality/illegality of abortions, as laid out in the *Abortion Debate* entry of the online Wikipedia [17]. The description is a summary of what appears in more detail elsewhere [18].

#### 3.1. Capturing representations of the debate in a knowledge base

Figure 3 (unshaded portion) shows an extract from the Wikipedia entry which expresses some of the debate issues. Figure 3 (shaded portion) then shows how the

first of the questions in the extract (underlined) is captured as an *Issue* instance in the knowledge base<sup>6</sup>. Note that this issue instance is asserted as a sub-issue of the main issue being debate – i.e. “*What should be the legal status of abortions*”.

<p><i>Some of the most significant and common issues treated in the abortion debate are:</i></p> <ul style="list-style-type: none"> <li>▪ <i>The beginning of personhood (sometimes phrased ambiguously as "the beginning of life"): When is the embryo or fetus considered a person?</i></li> <li>▪ <i>Universal human rights: Is aborting a zygote, embryo, or fetus a violation of human rights? ...</i></li> </ul> <p>...</p>
<pre>(def-instance ISS1 Issue   ((verbalExpression "What should be the legal status of abortions?")))  (def-instance ISS2 Issue   ((verbalExpression "When is the embryo or fetus considered a person?")    (subIssueOf ISS1))) ...</pre>

**Figure 3.** (Unshaded portion) An extract from the Wikipedia entry showing some of the debate issues (Shaded portion) ‘Issue’ instances modelled in the knowledge base.

Next the process turns to representing the claims and arguments in the debate. According to the Wikipedia entry, the argumentation in the debate is generated by two broadly opposing viewpoints – *anti-abortion* and *pro-abortion*. Figure 4 (unshaded portion) shows an extract that expresses three basic ‘anti-abortion’ claims. Figure 4 (shaded portion) then shows how the first of these three claims is captured as a *Proposition* instance. An *Argument* instance is then coded – *BASIC-ANTI-ABORTION-ARGUMENT* – which groups the anti-abortion claims together. Similar steps are performed to represent the basic pro-abortion viewpoint. Next, an ‘addresses’ link is asserted between the *BASIC-ANTI-ABORTION-ARGUMENT Argument* instance and the main debate issue. Finally, a ‘disputes’ link is asserted between the two *Argument* instances *BASIC-ANTI-ABORTION-ARGUMENT* and *BASIC-PRO-ABORTION-ARGUMENT*.

<p><i>The view that all or almost all abortion should be illegal generally rests on the claims: (1) that the existence and moral right to life of human beings (human organisms) begins at or near conception-fertilisation; (2) that induced abortion is the deliberate and unjust killing of the fetus in violation of its right to life; and (3) that the law should prohibit unjust violations of the right to life.</i></p> <p>...</p>
<pre>(def-instance P1 Proposition   ((verbalExpression "The existence and moral right to life of human organisms begins at or near conception-fertilisation"))) ... (def-instance BASIC-ANTI-ABORTION-ARGUMENT Argument   ((hasPremise P1 P2 P3)    (hasConclusion P4))) ... (def-relation-instances   (addresses BASIC-ANTI-ABORTION-ARGUMENT ISS1)   (disputes BASIC-ANTI-ABORTION-ARGUMENT BASIC-PRO-ABORTION-ARGUMENT)) ...</pre>

**Figure 4.** (Unshaded portion) An extract from the Wikipedia entry showing the basic anti-abortion viewpoint in the debate. (Shaded portion) Examples of claims and arguments modelled in the knowledge base.

<sup>6</sup> The knowledge base is coded using the OCML [19] knowledge modelling language.

The Wikipedia entry also includes information about publications and authors. Figure 5 (unshaded portion) shows an extract from the Wikipedia entry detailing the reference information for two publications by an author participating in the debate. Figure 5 (shaded portion) shows how the author is modelled as a Person instance and how the first publication is modelled as a Publication instance in the knowledge base. The shaded portion of the figure also shows how the fact that a publication ‘expresses’ an argument is captured in the knowledge base. In this case, the Thomson (1970) publication expresses an argument labelled as the ‘bodily-rights argument’, which supports the basic pro-abortion viewpoint in the debate.

<ul style="list-style-type: none"> <li>▪ Thomson, J. “A Defense of Abortion”. <i>Philosophy and Public Affairs</i> 1:1 (Autumn 1971): 47-66.</li> <li>▪ Thomson, J. “Rights and Deaths”. <i>Philosophy and Public Affairs</i> 2:2 (Winter 1973): 146-159.</li> </ul> <p>...</p> <pre>(def-instance JUDITH_THOMSON Person)  (def-instance THOMSON1971DEFENSE Publication   ((hasAuthor JUDITH_THOMSON)    (hasTitle "A defense of abortion")    (hasYear 1971)))  (def-relation-instances   (expresses THOMSON1971DEFENSE BODILY-RIGHTS-ARGUMENT)   (supports BODILY-RIGHTS-ARGUMENT BASIC-PRO-ABORTION-ARGUMENT))</pre>
---

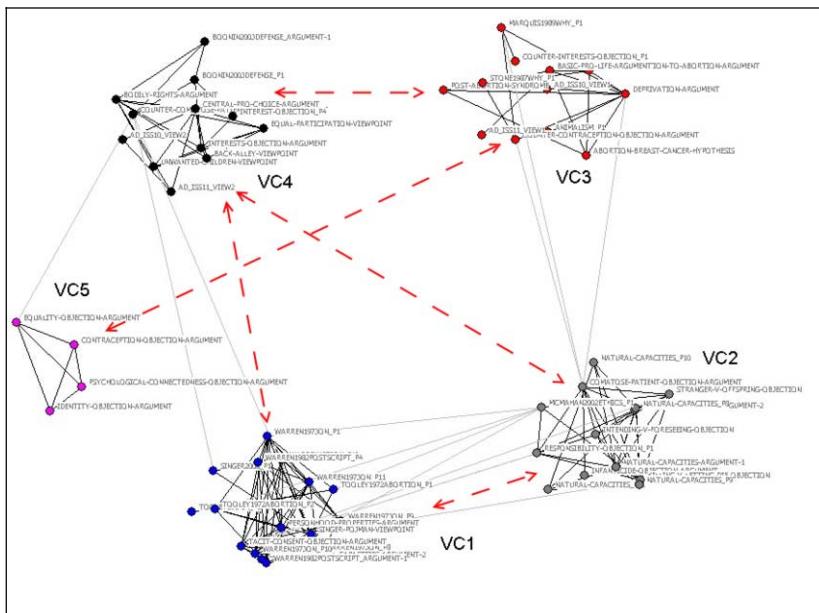
**Figure 5.** (Unshaded portion) An extract from the Wikipedia entry showing the reference information for two publications. (ii) (Shaded portion) The corresponding instances modelled in the knowledge base.

### 3.2. Detecting viewpoint clusters in the debate

The purpose of modelling a debate in a knowledge base is to enable new kinds of analytical services for detecting significant features of the debate. One such feature is the clustering of viewpoints in the debate into cohesive subgroups. A service to detect viewpoint clusters is of significance to a hypothetical end-user because understanding how scholars in a debate fall into different subgroups has already been established as an important part of understanding a debate [10].

The first step in executing this service is to translate the ontology-based representation into a single-link-type network representation so that the clustering technique reused from Bibliometrics can be applied. This involves executing the previously introduced inference rules on the entire ontology-based representation, which results in a network representation with a single +ADDITIVE link type. Next, a clustering algorithm is run over the network representation, which yields a number of clustering arrangements, ranging from 2 clusters to 25 clusters. The algorithm designers [20] propose a ‘goodness-of-fit- measure as an objective means of choosing the number of clusters into which the network should be divided. Figure 6 shows the clustering arrangement with five clusters, which the algorithm determines in this case is the arrangement with the ‘best fit’ for the given network data<sup>7</sup>. However, it should be noted that, as is typical in the Bibliometrics tradition (e.g. [21]), the goal of this type of analysis is typically not to find the definitive clustering arrangement, but rather to detect interesting phenomena that will motivate more focussed investigation on the part of the analyst.

<sup>7</sup> The Netdraw network analysis tool (available at <http://www.analytictech.com/Netdraw/netdraw.htm>) is used here to perform the clustering and visualisation.



**Figure 6.** Five viewpoint clusters (V.C. #1 – V.C. #5) identified in the abortion debate network. The dashed red lines between clusters indicate opposition (which is determined by further analysis once the clustering results are translated back into the knowledge base)

Once the viewpoint clusters have been identified the results are translated back into the knowledge base where each of the clusters detected in the network is captured as a *ViewpointCluster* instance. Further analysis is then performed on these instances to determine, for example, the persons who are associated with each cluster and the clusters which are deemed to be opposing each other. Two clusters are regarded as opposing if at least half of the viewpoints in one cluster have a ‘*disputes*’ relation with the viewpoints in the other cluster. Table 2 shows for each cluster, the associated viewpoints, the associated person(s), and the opposing cluster(s).

**Table 2.** Further details of the viewpoint clusters detected in the abortion debate, including the persons associated with each cluster and the clusters that are opposing each other.

VC#	Associated Viewpoints	Associated Person(s)	Opposing Cluster(s)
VC1	COUNTER-NATURAL-CAPACITIES-ARGUMENT <sup>8</sup> , PERSONHOOD-PROPERTIES-ARGUMENT <sup>9</sup>	Bonnie Steinbock, David Boonin, Dean Stretton, Jeff McMahan, Louis Pojman, Mary-Anne Warren, Michael Tooley, Peter Singer	VC2 VC4
VC2	COMATOSE-PATIENT-OBJECTION-ARGUMENT <sup>10</sup> ,	Don Marquis, Francis Beckwith, Germaine Grisez, Jeff McMahan, John Finnis, Katherine	VC1

<sup>8</sup> Summary: "The argument that the fetus itself will develop complex mental qualities fails."

<sup>9</sup> Summary: "The fetus is not a person because it has at most one of the properties – consciousness – that characterizes a person."

<sup>10</sup> Summary: "Personhood criteria are not a justifiable way to determine right to life since patients in reversible comas do not exhibit the criteria for personhood yet they still have a right to life"

	INFANTICIDE-OBJECTION-ARGUMENT <sup>11</sup>	Rogers, Massimo Reichlin, Patrick Lee, Robert George, Robert Larmer, Stephen Schwarz	VC4
VC3	ABORTION-BREAST-CANCER-HYPOTHESIS <sup>12</sup> , BASIC-ANTI-ABORTION-ARGUMENT	Don Marquis, Eric Olson, Jim Stone	VC4 VC5
VC4	BODILY-RIGHTS-ARGUMENT <sup>13</sup> , BASIC-PRO-ABORTION-ARGUMENT	David Boonin, Dean Stretton, Jonathan Glover, Judith Thomson, Peter Singer	VC1 VC2 VC3
VC5	CONTRACEPTION-OBJECTION-ARGUMENT <sup>14</sup> , IDENTITY-OBJECTION-ARGUMENT <sup>15</sup>	Dean Stretton, Frederick Doepke, Gerald Paske, Jeff McMahan, Lynne Baker, Mary-Anne Warren, Michael Tooley, Peter McInerney, William Hasker	VC3

## 4. Discussion

The discussion section is organised around a series of questions adapted from the GlobalArgument.net experiment<sup>16</sup>. These questions were used to evaluate various computer-supported argumentation (CSA) approaches to modelling the Iraq Debate. The discussion is concerned with two main points: the added value of the approach, and the limitations of the approach.

### 4.1. In what ways does this CSA approach add value?

*How does this CSA approach guide a reader/analyst through the debate?*

The aim of this approach is to provide analytical services that enable the reader to identify interesting features of the debate and gain insights that may not be readily obtained from the raw source material alone. As viewpoint clusters provide a way of abstracting from the complexity of the debate, an approach that enables the detection of viewpoint clusters in a debate is an improvement on what a user would have been able to obtain from looking only at the online Wikipedia entry of the abortion debate. Another interesting feature of the debate that the table reveals is that some persons are associated with more than one cluster. For example, the person of *Jeff McMahan* is associated with two clusters (VC1 and VC2), and furthermore these clusters happen to oppose each other. These are features of the debate that can then be investigated down to the level of the individual arguments.

<sup>11</sup> Summary: "Using personhood criteria would permit not only abortion but infanticide"

<sup>12</sup> Summary: "There is a causal relationship between induced abortion and an increased risk of developing breast cancer"

<sup>13</sup> Summary: "Abortion is in some circumstances permissible even if the fetus has a right to life because even if the fetus has a right to life, it does not have a right to use the pregnant woman's body."

<sup>14</sup> Summary: "It is unsound to argue that abortion is wrong because it deprives the fetus of a valuable future as this entails that contraception, which deprive sperm and ova of a future, is as wrong as murder, something which most people don't believe."

<sup>15</sup> Summary: "The fetus does not itself have a future value but has merely the potential to give rise to a different entity, an embodied mind or a person, that would have a future of value"

<sup>16</sup> <http://kmi.open.ac.uk/projects/GlobalArgument.net>

*To what extent is the modeller's or analyst's expertise critical to achieving the added value?*

Capturing the debate in the knowledge base often relied on the modeller's ability to reconstruct argumentation to include parts of arguments not expressed in the original information resource as well as inter-argument relations not expressed in the original information resource. This has an impact on what kinds of connections can be inferred during the reasoning steps, which then has an impact on what features of the debate can be identified.

#### *4.2. What are the limitations of the CSA approach?*

*What aspects of the debate proved difficult to model?*

Using this approach it was difficult to account for the different types of *disputes* relations between arguments. For example, in the case study, one argument often disputed another, not just because of disagreement with the conclusion, but because of the perceived 'unsoundness' of the reasoning used to arrive at the conclusion. Also, because of the focus on macro-argumentation, it was difficult to account for different inference schemes for moving from premises to conclusion in individual arguments.

*What missing capabilities have been identified?*

One important missing capability is (semi)automatically capturing debate representations from primary literature sources (e.g. experimental articles carried by scholarly journals). The approach relied on a manual process of constructing representations of the debate based on the Wikipedia, which is classified as a tertiary literature source [22]. Tertiary literature – which includes encyclopaedias, handbooks and review articles – consolidates and synthesises the primary literature thus providing an entry point into the particular domain. This was used for the case study to enable the manual coding of the debate, which would have been too vast to code using all the primary literature that was synthesised. However, this has meant that the debate representations rely on the accuracy of the tertiary-level synthesis of the primary literature.

## **5. Conclusion and Future Work**

This paper has described how an ontology of the structure of specialist knowledge domains can be extended to support the representation and analysis of scholarly debates within knowledge domains. The benefit of this extension has then been demonstrated by representing and analysing a case study debate. In particular, a service was demonstrated for detecting 'viewpoint clusters' as significant features of scholarly debate. Other debate modelling case studies are currently being written up.

Future directions for this research need to address the difficult question of how to best support the use of primary literature as the source for capturing debate representations. To cover a significant area of any knowledge domain, the modelling of primary literature would need to be conducted in a (semi)automated distributed fashion using possibly many modellers. This would introduce new challenges of trying to ensure consistent modelling across the different modellers. Finally, to address the current limitations of representing micro-argumentation, future research needs to

investigate how the ontology can incorporate the argument specification of the Argument Interchange Format (AIF) [9].

## References

- [1] Lawrence, S., C. Giles, and K. Bollacker, *Digital Libraries and Autonomous Citation Indexing*. IEEE Computer, 1999. **32**(6): p. 67-71.
- [2] Chen, C., *CiteSpace II: Detecting and Visualizing Emerging Trends and Transient Patterns in Scientific Literature*. Journal of the American Society for Information Science and Technology, 2006. **57**(3): p. 359–377.
- [3] Haase, P., et al., *Bibster - a semantics-based bibliographic Peer-to-Peer system*. Journal of Web Semantics, 2004. **2**(1): p. 99-103.
- [4] Kampa, S., *Who are the experts? E-Scholars in the Semantic Web*, in *Department of Electronics and Computer Science*. 2002, University of Southampton.
- [5] Schraefel, M.M.C., et al. *CS AKTive space: representing computer science in the semantic web*. in *13th International Conference on World Wide Web, WWW 2004*. 2004. New York, NY, USA: ACM.
- [6] Uren, V., et al., *Sensemaking tools for understanding research literatures: Design, implementation and user evaluation*. International Journal of Human-Computer Studies, 2006. **64**(5): p. 420-445.
- [7] Atkin, A., *Peirce's Theory of Signs*, in *The Stanford Encyclopedia of Philosophy (Fall 2007 Edition)*, E.N. Zalta, Editor. 2007, URL = <<http://plato.stanford.edu/archives/fall2007/entries/peirce-semiotics/>>.
- [8] Behrendt, W., et al. *Towards an Ontology-Based Distributed Architecture for Paid Content*. in *The Semantic Web: Research and Applications, 2nd European Semantic Web Conference (ESWC 2005)*. 2005. Heraklion, Crete, Greece: Springer-Verlag.
- [9] Chesñevar, C., et al., *Towards an argument interchange format*. Knowledge Engineering Review, 2006. **21**(4): p. 293-316.
- [10] Horn, R.E., *Mapping great debates: Can computers think? 7 maps and Handbook*. 1998, MacroVU: Bainbridge Island, WA.
- [11] Yoshimi, J., *Mapping the Structure of Debate*. Informal Logic, 2004. **24**(1): p. 1-21.
- [12] Kunz, W. and H.W.J. Rittel, *Issues as Elements of Information Systems*. 1970, Center for Planning and Development Research, University of California, Berkeley.
- [13] Walton, D., *Argumentation Schemes for Presumptive Reasoning*. 1996, Mahwah, New Jersey: Lawrence Erlbaum Associates.
- [14] Mancini, C. and S. Buckingham Shum, *Modelling discourse in contested domains: A semiotic and cognitive framework*. International Journal of Human-Computer Studies, 2006. **64**: p. 1154-1171.
- [15] Sanders, T.J.M., W.P.M. Spooren, and L.G.M. Noordman, *Towards a taxonomy of coherence relations*. Discourse Processes, 1992. **15**: p. 1-35.
- [16] Eemeren, F.H.v., et al., *Reconstructing Argumentative Discourse*. 1993, Tuscaloosa/London: The University of Alabama Press.
- [17] Website, *The Abortion Debate*, URL = <[http://en.wikipedia.org/wiki/Abortion\\_debate](http://en.wikipedia.org/wiki/Abortion_debate)> (accessed on 22 September 2006).
- [18] Benn, N., et al., *Designing the Ontological Foundations for Knowledge Domain Analysis Technology: An Interim Report*. 2008, Knowledge Media Institute, The Open University, UK. Technical Report KMI-08-02. Available at: <http://kmi.open.ac.uk/publications/pdf/kmi-08-02.pdf>.
- [19] Motta, E., *Reusable Components for Knowledge Modelling: Case Studies in Parametric Design Problem Solving*. Frontiers in Artificial Intelligence and Applications, ed. J.L.d.M. Breuker, R.; Ohsuga, S.; Swartout, W. Vol. 53. 1999, Amsterdam: IOS Press.
- [20] Newman, M.E.J., *Fast algorithm for detecting community structure in networks*. Physical Review E, 2004. **69**(066133).
- [21] Andrews, J.E., *An author co-citation analysis of medical informatics*. Journal of the Medical Library Association, 2003. **91**(1): p. 47-56.
- [22] Anderson, J., *The role of subject literature in scholarly communication: An interpretation based on social epistemology*. Journal of Documentation, 2002. **58**(4): p. 463-481.

# Investigating Stories in a Formal Dialogue Game

Floris BEX<sup>a</sup>, Henry PRAKKEN<sup>a,b</sup>

<sup>a</sup>Centre for Law & ICT, Faculty of Law, University of Groningen

<sup>b</sup>Department of Information and Computing Sciences, Utrecht University

**Abstract** In this paper we propose a formal dialogue game in which two players aim to determine the best explanation for a set of observations. By assuming an adversarial setting, we force the players to advance and improve their own explanations as well as criticize their opponent's explanations, thus hopefully preventing the well-known problem of 'tunnel vision'. A main novelty of our approach is that the game supports the combination of argumentation with abductive inference to the best explanation.

## 1. Introduction

In the literature, two main approaches to reasoning about factual issues in criminal cases are the story-based and the argument-based approach. Some authors [1] argue that legal reasoning about evidence is mainly done by constructing and analysing evidential arguments from the sources evidence to the events that are to be proven. Other authors [13] argue that legal reasoning with evidence is instead done by constructing different stories around the evidence and then analysing and comparing these stories. In previous work ([5], [6]), we have shown that these approaches can be combined in one formal framework, namely as a combination of the AI formalisms of abductive inference to the best explanation (IBE) and defeasible argumentation. Stories about what might have happened in a case are represented as hypothetical explanations and links between stories and the available evidence are expressed with evidential generalizations that express how parts of the explanations can be inferred from evidential sources with defeasible argumentation. Combining the argumentative and IBE approach in this way solves some of the problems of the separate approaches.

A limitation of our previous work is that it discusses only a static viewpoint: a framework is provided for the current status of an argumentative and story-based analysis of a case, and the dynamics of developing and refining such an analysis are not discussed. In this paper, we aim to model these dynamics in a formal dialogue game. In this dialogue game, it should be possible to build, critically analyse and change stories and their supporting arguments.

The game we propose is for dialogues in which crime analysts aim to determine the best explanation for a set of observations. Despite this cooperative goal of the dialogue participants, we still assume an adversarial setting in which the protocol is designed to motivate the players to 'win'. Thus we hope to prevent the well-known problem of 'tunnel vision' or confirmation bias, by forcing the participants to look at all sides of a case. A main novelty of our approach, motivated by our previous work in

[5] on the static aspects of crime investigation, is that the game supports the combination of argumentation with IBE.

The rest of this paper is organized as follows. In section 2, we summarize our combined framework as developed in [5]. In section 3 we present our formal dialogue game and in section 4 we apply it to a simple example. Finally, in section 5 we conclude with a discussion and some ideas for future research.

## 2. Argumentative Story-based Analysis of Reasoning with Evidence

In this section, the framework for argumentative story-based analysis of reasoning with evidence as proposed in [5] will be summarized. This formal framework combines a logic for defeasible argumentation with a logical model of abductive inference to the best explanation (for an overview of abductive reasoning see [10]). We first discuss our combined logical framework, followed by a short example. Then we will argue that this combined approach solves some of the problems of purely argumentative and purely IBE approaches to evidential reasoning.

The basic idea of the combined approach is as follows. A logical model of abductive IBE takes as input a causal theory and a set of propositions that has to be explained, the *explananda*, and produces as output a set of hypotheses that explain the explananda in terms of the causal theory. The combination of hypotheses and causal theory can be seen as a story about what might have happened. These hypothetical stories can then be compared according to the extent to which they conform to the evidence in a case. This evidence is connected to the stories by defeasible arguments from evidential sources (e.g. witness testimonies). Defeasible arguments are also used to reason about the plausibility of a story: the causal rules of the causal theory are not just given but their applicability can become the subject of an argumentation process. This definition of stories as causal networks is not entirely new: Pennington and Hastie [13] also defined stories as causal networks, following earlier influential research by Schank and Abelson [18]. Note that we use a naïve interpretation of causality; sometimes a causal link does not represent a much stronger relation than temporal precedence. This allows us to model a story as a simple, chronologically ordered sequence of events.

A framework for evidential reasoning  $ER = (C, A)$  is a combination of a causal-abductive framework  $C$  and an evidential argumentation framework  $A$ . The underlying logic  $L$  of this framework consists of the inference rules of classical logic combined with a defeasible modus ponens rule for a conditional operator  $\Rightarrow$  for defeasible generalizations. The generalizations used in  $C$  and  $A$  (see below) are formalized with this connective:  $g_i; p_1 \wedge \dots \wedge p_n \Rightarrow q$ . Here  $g_i$  is the name of the generalisation and  $p_1 \dots p_n$  and  $q$  are literals. The type of generalisation is indicated with a subscript:  $\Rightarrow_E$  denotes an evidential generalisation and  $\Rightarrow_C$  denotes a causal generalisation. For example,  $\text{Smoke} \Rightarrow_E \text{Fire}$  says that smoke is evidence of fire, while  $\text{Fire} \Rightarrow_C \text{Smoke}$  says that fire causes smoke.

The *argumentation framework* is a pair  $A = (G, I)$ , where  $G$  is a set of evidential generalizations and  $I$  is a set of input facts, where  $I_E \subseteq I$  is the set of sources of evidence in a case. This set of evidence  $I_E$  is different from other input facts in that the sources of evidence in  $I_E$  cannot be attacked by arguments. The other elements in  $I$  are propositions that denote ‘general knowledge’, opinions or ideas which may be open to

discussion and which are not generalizations of the form ‘if...then...’ (e.g. ‘Hillary Clinton will probably be the next U.S. president’ or ‘the idea that people from Suriname rob supermarkets more often than Dutch people is based on prejudice’).

The logic for this framework is very similar to the logic underlying the ASPIC inference engine [1], which in turn combines Pollock’s [14] ideas on a tree structure of arguments and two notions of rebutting and undercutting defeat with Prakken & Sartor’s [17] rule language and their argument game for Dung’s [8] grounded semantics. The set  $I$  and the evidential generalizations from  $G$  allow us to build evidential arguments by taking elements from  $I$  and the generalizations as premises and chaining applications of defeasible modus ponens into tree-structured arguments. Such an *evidential argument* is a finite sequence of lines of argument, where a line is either a proposition from  $I$ , a generalization from  $G$  or the result of an application of the defeasible modus ponens to one or more previous lines.  $\text{Args}(A)$  is the set of all well-formed arguments in  $A$ .

An argument can defeat another argument by rebutting or undercutting the other argument. Two arguments *rebut* each other if they have the opposite conclusion. An argument  $AR_1$  *undercuts* another argument  $AR_2$  if there is a line  $\neg g_i$  in argument  $AR_1$  and a line in argument  $AR_2$ , which is obtained from some previous lines in  $AR_2$  by the application of defeasible modus ponens to  $g_i$ .

For a collection of arguments and their binary defeat relations, the dialectical status of the arguments can be determined: arguments can be either *justified*, which means that they are not attacked by other justified arguments that are stronger, or *overruled*, which means that they are attacked by one or more other stronger arguments that are justified, or *defensible*, which means that they are neither justified nor overruled. Note that in the present paper, we will not discuss the relative strength between arguments.

The *abductive framework* is a tuple  $C = (H, T, F)$ . Here,  $T$  is the *causal theory* which contains all the causal generalizations from the different stories.  $H$  is a set of *hypotheses*, propositions with which we want to explain the explananda.  $F$  is the set of *explananda*, propositions that have to be explained. A set of hypotheses  $H$  and a causal theory  $T$  can be used to explain propositions:

**(explaining)**  $H_i \cup T_i$ , where  $H_i \subseteq H$  and  $T_i \subseteq T$ , *explains* a set of propositions  $E$  iff

1.  $\forall e : \text{If } e \in E \text{ then:}$ 
  - $H_i \cup T_i \vdash e$ ; and
  - $H_i \cup T_i$  is consistent.
2. There is no justified argument in  $\text{Args}(A)$  for the conclusion  $\neg g$ , where  $g \in T_i$ .

Here  $\vdash$  stands for logical consequence according to the set of all deductive inference rules extended with modus ponens for  $\Rightarrow$ . Condition (1) of this definition is standard in logical models of abduction but condition (2) is new and makes it possible to attack causal generalizations of dubious quality in the explanation with an argument:  $H_i \cup T_i$  does not explain  $E$  if one of the generalizations in  $T$  is attacked by a justified argument.

The idea of our dialogue game is that during a dialogue the players jointly and incrementally build a framework, which can contain several alternative explanations for the explananda. Moreover, these explanations can be extended during the dialogue, for instance, by giving a further explanation for a hypothesis. Therefore we must be able to identify at each step of the dialogue the explanations for  $F$ .

**(explanation)** Given a framework  $ER$ , an explanation  $S = H_i \cup T_i$  is an explanation for the explananda  $F$  iff:

1.  $S$  explains  $F$ ; and

2.  $H_i \subseteq H$  contains only initial causes; and
3.  $T_i$  is a minimal subset of  $T$ .

Initial causes are propositions that are not a conclusion of a causal rule in  $T$ . This ensures that an explanation is considered from beginning to end. The condition that  $T_i$  is a minimal subset of  $T$  ensures that two explanations for  $F$  are really seen as two different explanations. The set of all explanations for  $F$  in a framework is denoted as  $\text{Expl}(ER)$ . If there is more than one explanation for the explananda, they must be compared according to their plausibility and their conformity to the evidence in a case.

The plausibility of a story is often judged by looking at the plausibility of its underlying generalizations (cf. [19]). Two kinds of generalizations are important to consider: the causal generalizations in the explanation and the evidential generalizations in the arguments linking the evidence to the story. The plausibility of the causal generalizations is ensured by point 2 of the definition of explaining on the previous page. In the same way, the plausibility of the evidential generalizations in the arguments is ensured by allowing arguments to be attacked and defeated.

As to an explanation's conformity to the evidence in a case, we recognize three criteria. The first of these is *evidential coverage*, which stands for the number of sources of evidence covered by an explanation. The second is *evidential contradiction*, which stands for the number of events in the explanation contradicted by evidential arguments and the third is *evidential support*, which stands for the number of events in a story supported by evidential arguments. Evidential coverage was first mentioned in [13] and the other criteria were mentioned in [6]. Because in this paper the focus is on the dialogue game, we only formally define evidential coverage.

**(evidential coverage)** The *evidential coverage* of an explanation  $S$ , denoted as  $ec_S$ , is the total number of sources of evidence that are *covered* by an explanation, where:

- a source of evidence  $p \in I_E$  is covered by an explanation  $S$  if a proposition in  $S$  follows from a non-overruled argument in  $\text{Args}(A)$  which has  $p$  as its premise.

Thus, if a proposition in the explanation follows from a source of evidence the explanation covers that source of evidence. So if, for example, an explanation  $S$  covers five pieces of evidence, then  $ec_S = 5$ . Note that here we do not intend to define an objective probabilistic measure for the quality of stories; instead the notion of evidential coverage aids us in comparing explanations, viz.: an explanation  $S$  is better than an explanation  $S'$  if its evidential coverage is higher. Note how the above definition ensures the plausibility of the evidential generalizations: if an argument that links a certain piece of evidence to an explanation is overruled, that piece of evidence does not count towards the evidential coverage of an explanation.

The explananda, while they also follow from evidence and are also events in an explanation, are treated differently from other events in an explanation; explananda cannot be attacked and providing arguments from evidence for an explanandum does not increase an explanation's evidential coverage. This is because we do not want to reason about *what* should be explained but instead we want to reason about *how* certain events are explained. In section 3.2 this point will be made clearer.

Let us illustrate the combined framework with a simple example, adapted from Wagenaar et al. ([19], page 35). The example concerns the Haaknat case, in which a supermarket was robbed. The police conducted a search operation in a park near the supermarket, hoping to find the robber. Haaknat was found hiding in a moat in the park and the police, believing that Haaknat was the robber, apprehended him. Haaknat, however, argued that he was hiding in the moat because earlier that day, he had an

argument with a man called Benny over some money. According to Haaknat, Benny drew a knife so Haaknat fled and hid himself in the moat where the police found him. The explanandum in this case is ‘Haaknat is found by the police’. In figure 1 the two explanations for this explanandum are represented in a simple graph.

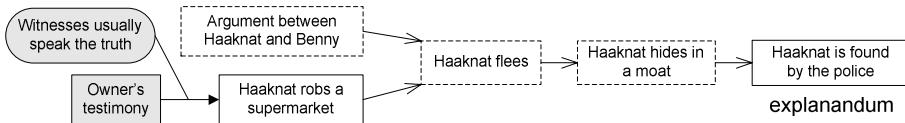


Figure 1: two explanations for the fact that Haaknat was found in the moat

In the figure, the causal theories combined with the hypotheses are represented as white boxes, where the variables in the causal theories have been instantiated with the constants from the hypotheses. Causal relations are rendered as arrows with an open head. A piece of evidence, namely, that the owner of the supermarket testified that it was Haaknat who robbed his shop, is represented as a grey box and the evidential generalization is represented as a grey rounded box; evidential relations are rendered as arrows with a closed head. In the example ‘Haaknat robs a supermarket’ follows from a non-overruled evidential argument. This can be seen in the figure, where events that are not supported by evidence are in a dotted box and events that are supported by evidence in a box with a solid line. The explanandum of course also follows from evidence (in this case a police report and Haaknat’s own testimony). However, the links between this evidence and the explanandum have not been rendered in the above figure, because we want to focus on whether the explanation is supported by evidence and not on whether the explanandum is supported by evidence. The bottom explanation (that Haaknat robbed the supermarket) can be regarded as the best explanation because it has an evidential coverage of 1 while the other explanation (that Haaknat had an argument with Benny) has an evidential coverage of 0.

Both the argumentative and the IBE approach have been used separately to model evidential reasoning (see [4] for an example of a purely argumentative approach and [19] for an example of an IBE approach). We now briefly explain why we combine the two approaches instead of adopting just one of them.

A disadvantage of an argumentative approach is that it does not provide a complete overview of the case, as the original stories about ‘what happened’ are cut into pieces to become conclusions of different arguments and counter-arguments. An approach where stories are represented as causal networks and thus the overview of the case is retained is closer to how legal decision makers and investigators actually think about a case ([13],[15]). This was informally confirmed in our contacts with police detectives and lecturers of the Dutch police academy, in which we learned that crime investigators often visualise time lines and causal structures to make sense of a body of evidence.

However, a problem of a purely IBE approach is that sources of evidence such as testimonies are modelled as effects caused by the event for which they serve as evidence. The advantage of adding evidential arguments to the IBE approach is that in the combined theory, reasoning with sources of evidence is arguably more natural: events are inferred from evidence using evidential generalizations. In our informal contacts with the Dutch police we found that this is how crime analysts usually connect the available evidence with their temporal and causal models of a case.

Another problem of the IBE approach as it is usually modelled is that it is impossible to reason *about* the causal generalizations used in an explanation. In legal settings this is a limitation since it is well-known that in criminal cases the quality of

the generalisations used by crime investigators or lawyers to build explanations cannot be taken for granted ([1], [19]).

### 3. The dialogue game

The analysis of stories and evidence in a case is a process; exactly how this process takes form depends on who performs the analysis and the specific legal context the analysis is performed in. In a decision-making context, for example, the defence is confronted with a complete story about what happened, namely the prosecution's story. Usually, this story is already supported by evidence and the defence will try to attack the prosecutor's evidential arguments (by arguing, for example, that a witness is not trustworthy) or the defence gives an alternative explanation (for example, that an alleged killer acted in self-defence). In an investigation context, however, things are different. Often, a team of criminal investigators is faced with some initial evidence and they construct several possible stories (or scenarios) and then try to find new evidence that supports or discredits these scenarios. During the investigation there is constant interaction between the scenarios and the evidence: a scenario provides a frame in which new evidence can be interpreted and, at the same time, new evidence is used to support or discredit a scenario or to extend a scenario [15].

In this paper, we aim to model the dynamics of the process of analysing stories and arguments in a formal dialogue game, with which it should be possible to build, critically analyse and change explanations and their supporting arguments.

Dialogue games formulate principles for coherent dialogue between two or more players, and this coherence depends on the goal of a dialogue. In our previous work on dialogue games [3], one of the players made a claim which he had to defend, while the other player's goal was to dispute this claim. The goal of the dialogue game was to resolve this difference of opinion in a fair and effective way. By contrast, the current dialogue game is meant to regulate a discussion between analysts in a criminal case. In such a setting the players have identical roles since they both want to find a plausible and evidentially well-supported explanation for the explananda. Moreover, none of the players really wants to win, since they have the joint goal to find the best explanation of the explananda. As explained in the introduction, our dialogue game is designed to promote this joint goal of the players by forcing them in an adversarial setting, where they technically have the aim to 'win', so that all sides of a case are explored. Accordingly, the game allows both players, given an initial body of evidence, to propose, criticise and defend alternative explanations for what happened. The idea behind enforcing such an adversarial setting is to avoid the well-known problem of 'tunnel-vision' or confirmation bias, where one explanation is taken as the right one and the investigation focuses on finding evidence that supports this explanation while dismissing evidence that contradicts this explanation. Note that while the game has two players, extending the dialogue game to accommodate for more players is easy, thus allowing our dialogue game to support discussions between groups of analysts.

Now, in a dialogue the players build a framework for evidential reasoning *ER* by performing speech acts from a communication language  $L_c$ . With these speech acts, explanations can be given for the explananda *F*, and arguments can be moved for supporting explanations or for attacking explanations or other arguments, thus continually updating the framework *ER*. One part of the dialogue game is a *protocol*, which specifies the allowed moves at a certain point in the dialogue. Such a protocol is

essentially a normative model for how the process of an analysis of evidence and explanations should take place.

The dialogue game also has *commitment rules*, which specify the effects of a speech act on the propositional commitments of the dialogue participants. For instance, explaining the explananda with an explanation commits the speaker to the explanation and retracting a previously moved argument removes this argument from the speaker's commitments. Commitments can be used to constrain the allowed moves, for example, to disallow moves that make the speaker's commitments inconsistent. They can also be used to define *termination* and *outcome* of a dialogue. Recall that the objective of the game is to find the best explanation for the explananda so the outcome of a dialogue is an explanation together with its supporting arguments and the dialogue terminates if both players are committed to the best explanation. In addition, for nonterminated dialogues a notion of the *current winner* can be defined; this is the adversarial element of the dialogue. The current winner is the player that is committed to the currently best explanation for the explananda. The notion is used to control turn taking, with a rule that a player is to move until he has succeeded in becoming the current winner (cf. [9]).

We now turn to the definitions of the elements of our dialogue game. Because of space limitations, the definitions will in some places be semiformal. Below,  $AR \in \text{Args}(A)$  and  $\varphi \in \text{wff}(L)$ , where  $L$  is the underlying logic of the framework (see section 2). Dialogues take place between two players,  $p_1$  and  $p_2$ . The variable  $a$  ranges over the players, so that if  $a$  is one player, then  $\bar{a}$  is the other player.

The communication language  $L_c$  consists of the following locutions or speech acts:

- *argue AR*. The speaker states an argument.
- *explain (E, S)*. The speaker provides an abductive explanation  $S = H \cup T$  for a set of propositions  $E$ .
- *concede  $\varphi$* . The speaker admits that proposition  $\varphi$  is the case.
- *retract  $\varphi$* . The speaker declares that he is not committed (any more) to  $\varphi$ .

The speech act *explain* is new while the other locutions are well-known from the literature. A *dialogue*  $d$  is now a sequence of utterances of locutions from  $L_c$ , where  $d_0$  denotes the empty dialogue. Each utterance is called a *move*. The speaker of a move  $m$  is denoted by  $s(m)$ .

### 3.1. Commitments

The players' commitments are influenced by the moves they do during a dialogue. At the start of a dialogue the commitments of both players consist of just the explananda from  $F$ . The set  $\text{Comms}_s$  denoting the commitments of the speaker  $s$  is updated during a dialogue as follows. When  $s$  moves an *argue AR* move, the premises of  $AR$  and conclusions of  $AR$  are added to  $\text{Comms}_s$ ; when  $s$  moves an *explain (E, S)* move, the elements from  $E$  and  $S$  are added to  $\text{Comms}_s$ ; when  $s$  moves a *concede  $\varphi$*  move,  $\varphi$  is added to  $\text{Comms}_s$  and when  $s$  moves a *retract  $\varphi$*  move,  $\varphi$  is deleted from  $\text{Comms}_s$ .

### 3.2. The framework in the dialogue protocol

Recall that in our setup the dialogue participants jointly build a framework for evidential reasoning  $ER$ . In this framework, the set explananda  $F$  is given and assumed nonempty and the players can update the framework by providing explanations or arguments using the speech acts. The set  $F$  does not change during the dialogue, so it

must be agreed upon before the dialogue starts. It is in theory possible to have an argumentative dialogue about what the explananda are. However, the purpose of the current dialogue is to find explanations for certain observations and to compare these explanations; a dialogue about what should be explained is a different kind of dialogue the details of which we leave for future research.

$ER(d) = ((H(d), T(d), F(d)), (G(d), I(d)))$  stands for the evidential reasoning framework after dialogue  $d$ . The elements of this framework also denote the elements after a certain dialogue  $d$ ; so  $H(d)$  is  $H$  after dialogue  $d$ ,  $T(d)$  is  $T$  after dialogue  $d$  etcetera. When the speaker  $s$  makes a move, the framework is updated as follows. When  $s$  moves an *argue AR* move, the generalizations in the argument  $AR$  are added to the set of evidential generalizations  $G$  and the other premises of  $AR$  are added to  $I$ . When  $s$  moves an *explain* ( $E, (H' \cup T')$ ) move, the hypotheses in  $H'$  are added to  $H$  and the causal generalizations in  $T'$  are added to  $T$ . When  $s$  moves a *retract*  $\varphi$  move and  $\varphi \notin Comms_s(d)$ , then  $\varphi$  is removed from its corresponding element in the framework.

### 3.3. Turn taking and winning

Before the protocol itself is defined, two related notions need to be defined, namely turn taking and winning. A player  $a$  is the *current winner* of a dialogue  $d$  if there is an explanation  $S$ ,  $S \in Expl(ER(d))$  and  $S \subseteq Comms(a)$ , and for each other explanation  $S'$ ,  $S' \in Expl(ER(d))$  and  $S' \neq S$ , it holds that  $ec_S > ec_{S'}$ . So if there is only one explanation for the explananda in the current framework, a player is the current winner if he is the only player committed to that explanation. If there are more explanations, a player is the winner if he is the only player committed to the explanation that has the highest evidential coverage. Note that this definition of the current winner also allows that there is no current winner, namely when both players are committed to explanations with equal evidential coverage.

With the notion of a current winner, a turn taking rule can be defined as follows: *Turn* is a function that for each dialogue returns the players-to-move, such that  $Turn(d_0) = p_1$ ,  $Turn(d, m) = \bar{a}$  if  $a$  currently wins  $d$ , else if there is no current winner and  $Turn(d) = a$  then  $Turn(d, m) = a$ . Thus it is always the losing player's turn and even if he makes such a move that the other player is no longer the winner, he still has to become the winner himself. This situation ensures that both players try to advance and defend their respective explanations as opposed to the situation where one player gives an explanation and the other player constantly attacks this one explanation.

### 3.4. The protocol

The protocol  $P$  specifies the allowed moves at each stage of a dialogue. Its formal definition is as follows. For all moves  $m$  and dialogues  $d$  it holds that  $m \in P(d)$  if and only if all of the following conditions are satisfied:

1.  $Turn(d) = s(m)$
2.  $m$  was not already moved in  $d$  by the same player
3.  $Comms_s(d, m) \not\vdash \perp$
4. If  $m$  is an *argue AR* move (where  $\varphi$  is  $AR$ 's conclusion), then  $\varphi \notin F$  and
  - either  $\varphi = \neg ge_i$  for some  $ge_i \in G(d)$  or  $\varphi$  is a negation of an element in  $(I / I_E)$
  - either  $\varphi = \neg gc_i$  for some  $gc_i \in T(d)$

- or  $\exists S, S \in \text{Expl}(ER(d)), S \subseteq \text{Comms}(s)$  and  $ec_S(d, m) > ec_S(d)$
- 5. If  $m$  is an *explain* ( $E, S$ ) move, then
  - $S$  explains  $E$  and  $E \cap F \neq \emptyset$
  - $S$  explains  $E$  and;  $\forall e \in E: H(d) \cup T(d) \vdash e$  and  $e$  is a literal
- 6. If  $m$  is a *concede*  $\varphi$  move, then  $\varphi$  is in an element of  $ER(d)$  and  $\varphi \notin \text{Comms}_s(d)$
- 7. If  $m$  is a *retract*  $\varphi$  move, then  $\varphi \notin F$  and  $\varphi \in \text{Comms}_s(d)$
- 8.  $\neg \exists S: S \in \text{Expl}(ER(d)), S \subseteq \text{Comms}_s(d), S \subseteq \text{Comms}_s(d)$  and for each other explanation  $S'$ ,  $S' \in \text{Expl}(ER(d))$  and  $S' \neq S$ , it holds that  $ec_{S'} > ec_S$ .

The first two conditions say that only the player-to-move can make allowed moves and that a player may not repeat his moves. Condition (3) regulates the players' logical consistency. The first point of condition (4) states that an argument may be moved if it attacks another argument, that is, it attacks an evidential generalization or it attacks an element from input that is not a source of evidence. The second point states that an argument may be moved if it attacks an explanation, that is, it attacks a causal generalization. The third point states that an argument may be moved if it improves the evidential coverage of an explanation to which the player is committed. Condition (5) states that an *explain* move may be done if it explains (see definition on page 3) an explanandum, or if it explains a literal that follows from the current hypotheses and the current theory. Condition (6) ensures that a player concedes a proposition only if it is in the current framework and the player is not already committed to it. Condition (7) says that a player can only retract a proposition to which he is committed. Finally, condition (8) implies that a dialogue terminates if both players are fully committed to the best explanation.

In the current dialogue game, the legality of moves is defined in terms of the current framework for evidential reasoning: every move must be a sensible operation on  $ER$ . This way of defining the relevance of moves is different from, for example, [12], where relevance is enforced by a strict protocol and [16], where relevance is enforced by the reply structure on the speech acts in the communication language.

## 4. Example

For the example we return to the Haaknat case on page 5. The set of explananda  $F$  in this case is  $\{\text{Haaknat is found by police}\}$ . Player  $p_1$  starts the dialogue by providing an explanation for this explanandum:

$p_1: \text{explain } (\{\text{Haaknat is found by police}\}, \{\text{Haaknat robs supermarket}\} \cup T_1)$   
     where  $T_1 = \{\mathbf{gc}_1: x \text{ robs supermarket} \Rightarrow_C x \text{ flees}, \mathbf{gc}_2: x \text{ flees} \Rightarrow_C x \text{ hides in moat},$   
      $\mathbf{gc}_3: x \text{ hides in moat} \Rightarrow_C x \text{ is found by police}\}$

Now  $p_1$  is winning, because he is committed to the one explanation for  $F$ , which is obviously the best explanation.  $p_2$  at this point only has one option if he wants to become the current winner: he has to provide an explanation for  $F$  which is better than  $p_1$ 's explanation.

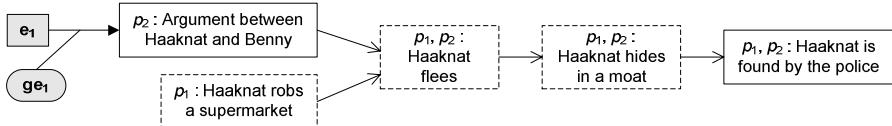
$p_2: \text{explain } (\{\text{Haaknat is found by police}\}, \{\text{argument between Haaknat and Benny}\} \cup T_2)$   
     where  $T_2 = \{\mathbf{gc}_4: \text{argument between } x \text{ and } y \Rightarrow_C x \text{ flees}, \mathbf{gc}_2, \mathbf{gc}_3\}$

After providing this explanation, it is still  $p_2$ 's turn, as the explanation he has provided is not better than  $p_1$ 's explanation.  $p_2$  supports his explanation by providing an argument. Below,  $\gg$  stands for the application of the defeasible modus ponens.

$p_2$ : argue  $AR_1$ :

- (e<sub>1</sub>: Haaknat's testimony "I had an argument with Benny"  $\wedge$  ge<sub>1</sub>: witness testifies that " $p$ "  $\Rightarrow_E p$ )  
 $\gg$  argument between Haaknat and Benny

Now  $p_2$  is the current winner: there is one piece of evidence in the case and it is covered by  $p_2$ 's explanation. The current framework is pictured in the figure below (the different arrows and boxes are explained on page 5). For each event, it is indicated which players are committed to that event.



At this point  $p_1$  can, for example, provide an argument for  $\neg \text{gc}_4$ ; if he does this, then  $p_2$  no longer has an explanation for  $F$  so  $p_1$  automatically has the best explanation. He can also try to support "Haaknat robs a supermarket" with at least two pieces of evidence, as this would make  $p_1$ 's explanation have a higher evidential coverage. Another option is to decrease the evidential coverage of  $p_2$ 's explanation by defeating the argument  $AR_1$ . Suppose that  $p_1$  chooses to take this last option:

$p_1$ : argue  $AR_2$ :

- (e<sub>2</sub>: Haaknat is a suspect in the case  $\wedge$  ge<sub>2</sub>: suspects do not make reliable witnesses)  $\gg \neg \text{ge}_1$

For the sake of the example, assume that this argument defeats  $AR_1$ .  $p_1$  is still not the current winner: both explanations have an evidential coverage of 0, so  $p_1$  has to make another move in order to make his explanation better or  $p_2$ 's explanation worse.  $p_1$  could increase the evidential contradiction of  $p_2$ 's explanation by providing an argument for  $\neg(\text{argument between Haaknat and Benny})$ . However, in this case  $p_1$  chooses to increase the evidential coverage of his own explanation. He does this by first expanding the explanation and then supporting it with evidence:

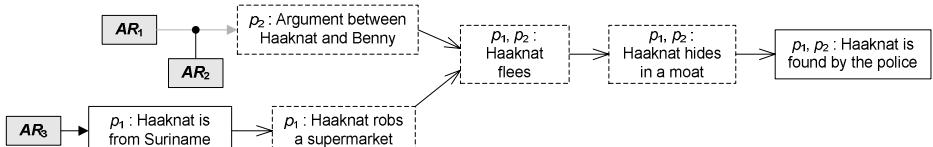
$p_1$ : explain ( $\{\text{Haaknat robs supermarket}\}$ ,  $\{\text{Haaknat is from Suriname}\} \cup T_3$   
 where  $T_3 = \{\text{gc}_5: x \text{ is from Suriname} \Rightarrow_C x \text{ robs supermarkets}\}$

$p_1$ : argue  $AR_3$ :

- (e<sub>1</sub>: Haaknat's birthplace is "Republic of Suriname"  $\wedge$

ge<sub>2</sub>:  $x$  birthplace is "Republic of Suriname"  $\Rightarrow_E x$  is from Suriname)  $\gg$  Haaknat is from Suriname

The following picture represents the current situation. The light grey argumentation arrow means that the inference is defeated and the arrow connected to  $AR_2$  stands for a defeat relation.

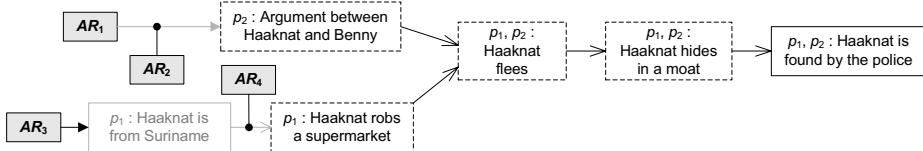


$p_1$  is now the winner: his explanation has an evidential coverage of 1 while  $p_2$ 's evidential coverage is 0. However, part of  $p_1$ 's generalization is based on the

generalization “people from Suriname rob supermarkets”.  $p_2$  does not agree with this and he argues that the generalization is based on prejudice:

$p_2$ : argue  $AR_4$ : ( $i_1$ :  $\text{gc}_5$  is based on prejudice,  $ge_2$ :  $\text{gc}_1$  is based on prejudice  $\Rightarrow_E \neg\text{gc}_1$ )  $\gg \neg\text{gc}_5$

Note that  $AR_4$  is not based on evidence, so it is possible for  $p_1$  to attack  $i_1$ . By attacking  $\text{gc}_5$ ,  $p_2$  ensures that  $\{\text{Haaknat is from Suriname}\} \cup T_1 \cup T_3$  is no longer an explanation. This is shown in the following figure, where  $AR_4$  attacks the causal generalization and the part of  $p_2$ 's explanation that is no longer considered is rendered in light grey.



$p_2$ 's explanation  $\{\text{argument between Haaknat and Benny}\} \cup T_2$  and  $p_1$ 's explanation  $\{\text{Haaknat robs supermarket}\} \cup T_1$  now both have an evidential coverage of 0, so  $p_2$  needs to make another move in order to become the winner.

## 5. Conclusions

In this paper we have shown how the combined story and argumentative approach to reasoning with evidence can be fit into a formal dialogue game. This dialogue game not only allows for the construction of defeasible arguments but also allows the players to explain events using abductive explanations, which can then be compared according to their conformity with the evidence in the case. Furthermore, the argumentative part of the game and framework allows players to have critical discussions about the plausibility and validity of the causal and evidential generalizations. The winning and turn taking conditions ensure that the players are forced to advance and improve their own explanations as well as criticize their opponent's explanations, which hopefully avoids ‘tunnel-vision’ or confirmation bias.

Our dialogue game can be seen as a guideline for a critical discussion between investigators in a criminal case. Furthermore, we intend to use the ideas presented in this paper in the further development of our software for police investigators, *AVERs*, which is currently being developed by other members of our project [7].

The precise form of our dialogue game is, to our knowledge, new. As far as we know, there are only two other dialogue games that allow the players to jointly build a theory ([11],[3]) build a Bayesian network and an argumentation-graph, respectively, whereas in our game a combined framework for argumentation and IBE is built. Our dialogue is a combination of an enquiry and a persuasion dialogue [21]; on the one hand, the players have a shared ‘quest for knowledge’ but on the other hand, the players try to persuade each other that their explanation is the best.

## Acknowledgements

This research was supported by the Netherlands Organisation for Scientific Research (NWO) under project number 634.000.429. The authors thank Bart Verheij for his useful comments on the ideas reported in this paper.

## References

- [1] L. Amgoud, L. Bodenstaff, M. Caminada, P. McBurney, S. Parsons, H. Prakken, J. van Veenen, G. Vreeswijk, Final review and report on formal argumentation system. ASPIC Deliverable D2.6, 2006
- [2] T.J. Anderson, D.A. Schum and W.L. Twining. *Analysis of Evidence, 2<sup>nd</sup> edition*. Cambridge University Press, 2005.
- [3] F.J. Bex and H. Prakken. Reinterpreting arguments in dialogue: an application to evidential reasoning. *Proceedings of JURIX 2004*, pp.119 – 129. IOS Press, 2004.
- [4] F.J. Bex, H. Prakken, C. Reed and D. Walton, Towards a formal account of reasoning about evidence: argumentation schemes and generalisations. *Artificial Intelligence and Law* 11, pp. 125 – 165, 2003.
- [5] F.J. Bex, H. Prakken and B. Verheij. Formalizing argumentative story-based analysis of evidence. *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ACM Press, 2007.
- [6] F.J. Bex, S.W. van den Braak, H. van Oostendorp, H. Prakken, H.B. Verheij & G.A.W. Vreeswijk Sense-making software for crime investigation: how to combine stories and arguments? *To appear in Law, Probability & Risk*, Oxford University Press, 2008
- [7] S.W. van den Braak, G.A.W. Vreeswijk, and H. Prakken, AVERs: An argument visualization tool for representing stories about evidence. *Proceedings of the 11th International Conference on Artificial Intelligence and Law*. ACM Press, 2007.
- [8] P.M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence* 77, pp. 321-357, 1995.
- [9] R.P. Loui. Process and policy: resource-bounded non-demonstrative reasoning. *Computational Intelligence*, 14:1-38, 1998.
- [10] P. Lucas, Symbolic diagnosis and its formalisation. *The Knowledge Engineering Review* 12, pp. 109 – 146, 1997
- [11] S.H. Nielsen and S. Parsons, An Application of Formal Argumentation: Fusing Bayes Nets in MAS. *Computational Models of Argument: Proceedings of COMMA 2006*, IOS Press, 2006
- [12] S. Parsons, M. Wooldridge and L. Amgoud, An analysis of formal inter-agent dialogues *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 394-401, ACM Press, 2002.
- [13] N. Pennington and R. Hastie. The story model for juror decision making. In R. Hastie (eds.), *Inside the Juror, The Psychology of Juror Decision Making*. Cambridge University Press, 1993.
- [14] J. Pollock, *Cognitive Carpentry: A Blueprint for How to Build a Person*, MIT Press, 1995.
- [15] C.J. de Poot, R.J. Bokhorst, P.J. van Koppen and E.R. Muller, *Recherchéportret - Over dilemma's in de opsporing (Investigation portrait – about dilemmas in investigation)*, Kluwer, 2004.
- [16] H. Prakken, Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation* 15, pp. 1009-1040, 2005.
- [17] H. Prakken and G. Sartor, Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics* 7, pp.25 – 75, 1997.
- [18] Schank, R. C. and Abelson, R. P. (1977) *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*, Lawrence Erlbaum, Hillsdale, NJ.
- [19] P. Thagard, Causal inference in legal decision making: Explanatory coherence vs. Bayesian networks. *Applied Artificial Intelligence*, 18, 231-249, 2004.
- [20] W.A. Wagenaar, P.J. van Koppen and H.F.M. Crombag, *Anchored Narratives. The Psychology of Criminal Evidence*. Harvester Wheatsheaf, 1993.
- [21] D.N. Walton and E.C.W. Krabbe, *Commitment in Dialogue. Basic Concepts of Interpersonal Reasoning*. State University of New York Press, 1995.

# Modeling Persuasiveness: change of uncertainty through agents' interactions

Katarzyna BUDZYŃSKA<sup>a</sup>, Magdalena KACPRZAK<sup>b,1</sup> and Paweł REMBELSKI<sup>c</sup>

<sup>a</sup> Institute of Philosophy, Cardinal Stefan Wyszyński University in Warsaw, Poland

<sup>b</sup> Faculty of Computer Science, Białystok University of Technology, Poland

<sup>c</sup> Faculty of Computer Science, Polish-Japanese Institute of Information Technology

**Abstract.** The purpose of this paper is to provide a formal model of a persuasion process in which a persuader tries to influence audience's beliefs. We focus on various interactions amongst agents rather than only exchange of arguments by verbal means. Moreover, in our approach the impact of the parties of the dispute on the final result of the persuasion is emphasized. Next, we present how to formalize this model using the modal system  $\mathcal{AG}_n$  inspired by Logic of Graded Modalities and Algorithmic and Dynamic Logics. We also show how to investigate local and global properties of multi-agent systems which can be expressed in the language of the proposed logic.

**Keywords.** effects of convincing, influence on uncertainty, grades of beliefs, nonverbal arguments, interactions amongst agents, modal semantics

## Introduction

The problem of understanding and formalizing argumentation as well as the issue of reaching agreement through argumentation have been studied by many researchers in different fields. A great deal of papers is devoted to study the theory, architecture and development of **argumentation-based systems**. The focus of most of these works is on the structure of arguments, their exchange as well as generation and evaluation of different types of arguments [2,5,10,13,15]. Much work has been done to analyze dialogue systems. For example P. M. Dung in [5] develops a theory that applies for argumentation which central notion is the acceptability of arguments. Moreover, he shows its connections to nonmonotonic reasoning and logic programming. In [10] a formal logic that forms a basis for a formal axiomatization system for argumentation's development is proposed. Furthermore, a logical model of the mental states of the agents based on a representation of their beliefs, desires, intentions and goals is introduced. Finally, a general Automated Negotiation Agent is developed and implemented.

An excellent review of **formal dialogue systems for persuasion** is given in [16]. The key elements of such systems are concerned with establishing *protocols* specifying allowed moves at each point in a dialogue, *effect rules* specifying the effects of utterances

---

<sup>1</sup>The author acknowledges support from Ministry of Science and Higher Education under Białystok University of Technology (grant W/WI/3/07).

on the participants' commitments, and *outcome rules* defining the outcome of a dialogue. Examples of frameworks for specifying persuasion dialogues are presented in [1,14,15]. In these approaches argument-based logics that conform Dung's grounded semantics are explored. They are used to verify whether agents' arguments are valid.

In our work we focus on persuasion rather than argumentation. Following Walton and Krabbe [19] we assume that the aim of the persuasion process is to resolve a conflict of opinion and thereby influence agents' beliefs. However, our purpose is not to create a novel persuasion (argumentation) theory. Instead, we provide a logical system which we are going to use to **investigate properties** of persuasion systems based on existing theories. Such a verification is assumed to be done formally, not experimentally. Therefore, our aim is not to develop and implement arguing agents or determine their architecture and specification. We introduce formal model of multi-agent systems and a modal logic which language is interpreted in this model. On this base we plan to test validity or satisfiability of formulas expressing specification of arguing agents as well as local and global properties of specific systems i.e. systems that can be expressed via our formal model. In current work we concentrate mainly on the formal model and the logic investigating their usefulness to description of the persuasion process.

Furthermore, we emphasize the impact which proponent and audience have on a persuasion process and its success. The real-life practice shows that this is not only arguments that affect our beliefs, but often people care more about who gives them. Thus, we are interested in including and examining the **subjective aspects** of persuasion in our formal model. This means that we focus on interactions among agents rather than only arguments. Those interactions contribute to success or failure of a persuasion process. We also stress that the influence on beliefs does not have to be accomplished by verbal means. Our paper provides a logical framework for handling not only persuasion dialogues but also various **nonverbal actions**.

Finally, we are interested in studying the course of persuasion step by step, i.e. how agents' beliefs about a given thesis are changed under an influence of particular arguments. It is the reason why we propose to use **grades of beliefs**, which are not explored in the approaches cited above. We want to be able to track how the successive arguments modify the degree of uncertainty of agent's beliefs at each stage of the persuasion (after the first argument, after the second, etc.). Therefore, we introduce the notion of **persuasiveness** understood as the degree of the audience's belief generated by the persuasion. It should be noted here that persuasiveness may also have other interpretations. It can be understood as success chances. The probabilistic approach to modeling such a notion is studied in [17]. The other interpretation can be found in [9] where persuasiveness is related to confidence in the recommendations provided by advising systems.

The paper is organized as follows. Section 1 explores the dynamics of persuasion, putting aside what logic should we choose to describe it. This is the goal of Section 2 where we propose the formal representation of two basic aspects of the persuasion dynamics: the grades of beliefs and their changes. We do not focus here on how the scenario of persuasion relates to its participants. This is the main topic of Section 3 where we demonstrate how effects of the persuasion differ depending on parties of dispute. Section 4 gives details about syntax and semantics of  $\mathcal{AG}_n$  logic. Section 5 shows how it can be used to investigate properties of multi-agent systems concerning a persuasion process.

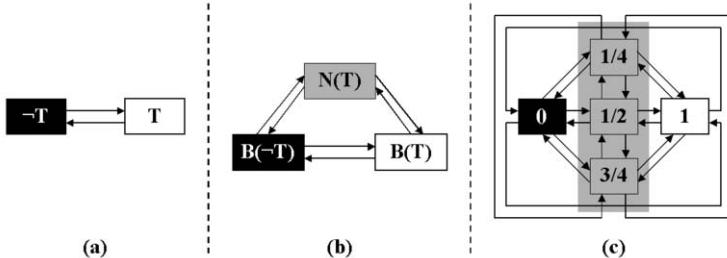
## 1. Dynamics of persuasion

The aim of our research is to investigate how a particular persuasion process influences the belief state of an agent. We use the notion of **persuasiveness** to describe this influence. More specifically, we say that the strength of a persuasion (i.e. of a given proponent and a sequence of arguments with respect to an audience and a thesis) is a degree of the audience's belief in the thesis which is generated by the sequence of arguments and the proponent. Thus, the stronger the confidence in the thesis after the persuasion, the more persuasive process of convincing. Such notion allows us to track the **dynamics** of a persuasion, i.e. the influence of this process on the beliefs of its audience. Let us consider a simple example of a persuasion. We will use it throughout the paper to illustrate the main ideas of our approach.

**(Example)** A young businesswoman plans a summer vacation and thinks of going somewhere warm - Italy, Spain or Mexico. The aim of a travel agent is incompatible with hers - he is going to sell her a vacation in Alaska. The travel agent starts a persuasion: "Maybe you would like to take a look on our last minutes offers? We have really good discounts for vacation to Alaska - you may compare vacation packages of a similar quality. We are not beaten on value and price" (**argument**  $a_1$ ). The customer replies: "Alaska - I don't really know. But it sounds very interesting". The travel agent continues using another tactic: "You seem to be a very creative person. Alaska has been very popular lately among people who are looking for unconventional places to visit" (**argument**  $a_2$ ). The businesswoman starts to be more interested: "It makes me think of Mexico, but Alaska sounds just as exciting". The travel agent shows a color leaflet about Alaska adventure including wildlife, bear viewing, kayaking etc. (**argument**  $a_3$ ). After a while, the woman says: "This is such a beautiful place! I will go there!".

What can we say about this persuasion? First of all, it was successful, i.e. the travel agent convinced the woman to go on vacation to Alaska. More specifically: (a) at the beginning she did not believe that she should spend a vacation in Alaska, (b) then the proponent executed three arguments  $a_1, a_2, a_3$ , (c) as a result, the woman became convinced to his thesis. Observe that the travel agent needed not one, but three arguments to achieve his goal. Say that we want to know what was happening with the businesswoman's beliefs between the beginning and the end of the persuasion, between her negative and positive belief state? If the travel agent did not stop after executing  $a_1$ , it can mean that at that moment she did not reach the absolutely positive belief attitude towards his thesis. We may describe this situation in terms of *uncertainty*, i.e., we can say that after  $a_1$  she was not absolutely certain she should spend her vacation in Alaska. In this sense, the persuasiveness is related to an influence on uncertainty of an audience's beliefs.

Let us have a closer look at the possibility of describing the dynamics of the persuasion process in different formal models. We will compare their expressibility using our example. Taking **propositional logic**, we can interpret a formula  $T$  as a belief of an agent. It gives us two modules representing her belief states (see Figure 1a): the box " $\neg T$ " means a negative belief "I don't believe a thesis  $T$ ", and the box " $T$ " - a positive belief "I do believe  $T$ ". It corresponds to this logic's tautology that for every sentence  $p$  it holds  $p \vee \neg p$ . In such a model, there is a possibility of two moves: from negative to positive belief or from positive to negative one. As a result, we can only express that in our example the persuasion consisting of the sequence  $a_1, a_2, a_3$  moves from the negative to the positive belief state of the businesswoman. However, we are not able to track what



**Figure 1.** The models of beliefs' changes induced by a persuasion: a) changes of a black-and-white type between negative and positive beliefs, b) changes of a black-gray-white type between negative, neutral and positive beliefs, c) changes with extended area of "grayness" representing different shades of uncertainty.

was happening with her beliefs at any intermediate stage of the process of convincing (for instance, after executing the argument  $a_1$ ). Notice that in the graph (a) in Figure 1, all three arguments have to be placed on one edge (i.e., the edge going from  $\neg T$  to  $T$ ).

To extend this black-and-white description of convincing, a **doxastic modal logic** may be used. It gives the opportunity of adding uncertainty to a model by introducing one module. Thus, we have the following boxes (see Figure 1b): " $B(\neg T)$ " which encodes a negative belief "I believe  $T$  is false", " $N(T)$ " - a neutral belief "I am not sure if  $T$  is true or false", and " $B(T)$ " - a positive belief "I believe  $T$  is true". It is strictly connected with the principle that not for every sentence  $p$ :  $Bp \vee B(\neg p)$  holds in this logic. Although the model extends the expressibility to six types of changes of beliefs, it still has a serious disadvantage. Observe that all various shades of uncertainty are put into one bag called "neutral belief". This means that when the arguments changes the audience's beliefs between greater and less uncertainty, such an oscillation moves only inside the gray (neutral) module. Consequently, those modifications cannot be distinguished within this framework. As we see, the phenomenon of persuasiveness has very limited possibilities to be expressed here. These limitations are easy to observe in our example. Let us assume that after each argument the woman's certainty raises and she becomes absolutely sure that she should spend her vacation in Alaska after execution of the argument  $a_3$ . How can these changes be illustrated in three-node model? Observe that if the persuasion is to move woman's beliefs from the negative state to the positive one, there are two edges at our disposal (the first one from  $B(\neg T)$  to  $N(T)$ , and the second one - from  $N(T)$  to  $B(T)$ ). Thus, two arguments must be put at the same edge. That is, in the graph (b) of Figure 1 the sequence  $a_1, a_2$  should be placed on one edge from  $B(\neg T)$  to  $N(T)$  (as long as we assume that she was absolutely persuaded of  $T$  not sooner than after executing  $a_3$ ). As a result, we are unable to express here what was happening with her belief state between execution of the argument  $a_1$  and execution of  $a_2$ .

The key to solve the problem is to design a model which enables to expand the area of grayness by adding so many modules as we need for a specific situation that we want to describe. These extra boxes are to represent different **shades of uncertainty**, and the transitions amongst them - the potential (even very subtle) movements that persuasion may bring about. For example, if we wanted to describe three types of uncertainty, our model should include five nodes: absolutely negative beliefs represented by  $0$ , rather negative  $\frac{1}{4}$ , fifty-fifty  $\frac{1}{2}$  (an agent thinks that chances of a thesis being true are about fifty-fifty), rather positive  $\frac{3}{4}$ , and absolutely positive  $1$  (see Figure 1c). Now, up to 20

changes which a persuasion may cause are allowed. As we mentioned, a choice of a number of these modules may be elastically and arbitrarily increased adapting the model to the particular applications. Let us show it in our example. Say that after executing the argument  $a_1$  the businesswoman is not absolutely negative about going to Alaska, but still rather sure she should not do it. Further, after  $a_2$  her certainty raises to the neutral attitude. In the graph 1c, we are able to illustrate the course of this persuasion: the argument  $a_1$  is placed on the edge from 0 to  $\frac{1}{4}$ ,  $a_2$  - on the edge from  $\frac{1}{4}$  to  $\frac{1}{2}$  and  $a_3$  - on the edge from  $\frac{1}{2}$  to 1. Consequently, we can track the dynamics of the persuasion observing how the audience reacts on each argument of the proponent, i.e., how her beliefs are modified by all these three arguments at any intermediate stage of the process of convincing. The issue how to formalize the degrees of beliefs and their changes is investigated in the next section.

## 2. Influence on uncertainty of beliefs

In this section we explain how we want to model the persuasiveness and we show the main ideas of the logic  $\mathcal{AG}_n$ . You will find the detailed presentation of this system in Section 4. The proposed logic adapts two frameworks combining them together. For representing uncertain beliefs of participants of the persuasion we use **Logic of Graded Modalities** (LGM) by van der Hoek and Meyer [18]. The great strength of this approach lies in its similarity to non-graded frameworks of representing cognitive attitudes, commonly used in AI and computer science. That is, LGM's language as well as axiomatization are the natural extensions of the modal system S5.<sup>2</sup> For representing beliefs' changes induced by convincing we apply the elements of a logic of programs like **Algorithmic Logic** (AL) [12] or **Dynamic Logic** (DL) [8]. A basic formula of  $\mathcal{AG}_n$  describing *uncertainty* is:  $M!_j^{d_1, d_2} T$  (where  $d_1, d_2$  are natural numbers) which intended reading is the following: in an agent  $j$ 's opinion a thesis  $T$  is true in exactly  $d_1$ -cases for  $d_2$ -doxastic alternatives.<sup>3</sup> We say that  $j$  believes  $T$  with the degree of  $\frac{d_1}{d_2}$ . A basic formula describing *change* is  $\diamond(i : P)M!_j^{d_1, d_2} T$  which intended reading is: after the sequence of arguments  $P$  performed by  $i$  it is possible that an agent  $j$  will believe  $T$  in a degree  $\frac{d_1}{d_2}$ .

This needs some explanation. First, we show the meaning of a formula  $M!_j^{d_1, d_2} T$ . In our example, at the beginning the businesswoman, now symbolized as *aud*, thinks the best place to spend a vacation is Italy, Spain or Mexico. Moreover, she imagines the reality in different ways, say in three ways for simplicity. The reality and *aud*'s visions of the reality are in Kripke-style semantics interpreted as *states (possible worlds)*. In Figure 2 the first module on the left describes the moment before the persuasion. The state  $s_1$  represents the reality,  $s_5$  is the state in which Italy is preferred as the best option for summer,  $s_6$  - the option of Spain,  $s_7$  - Mexico and  $s_8$  - the option of Alaska. The *accessibility (doxastic) relation*  $RB_{aud}$  is graphically illustrated as arrows connects a state with states that *aud* considers as its *doxastic alternatives*. In the example, the doxastic alternatives ( $s_5, s_6, s_7$ ) show what states are allowed by *aud* as the possible

<sup>2</sup>See [4] for the detailed discussion about pros and cons of this one and the other approaches which can be used to model graded beliefs. Moreover, see e.g. [11] for the overview of modal logics.

<sup>3</sup>This formula was added to LGM's language. Our goal was to make it more appropriate to the needs of expressing persuasiveness.

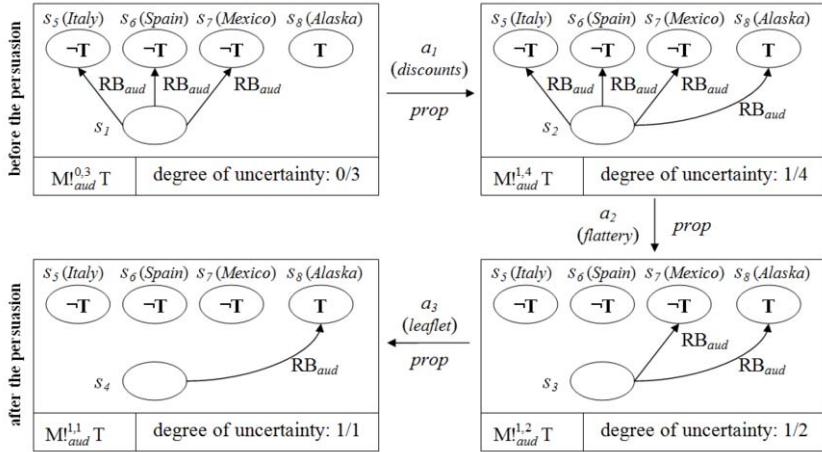


Figure 2. The change of the woman's uncertainty about the best place for summer during the persuasion.

visions of the reality ( $s_1$ ). Observe that before the persuasion  $aud$  excludes the possibility of going to Alaska what corresponds to the absence of connection between  $s_1$  and  $s_8$ . Let  $T$  stand for the thesis: "I am going on vacation to Alaska this summer". Clearly, it is false in  $s_5 - s_7$  and true in  $s_8$ . In the state  $s_1$  an agent  $aud$ 's **belief** on the thesis is represented by a formula  $M!_{aud}^{0,3} T$  since  $aud$  allows 0 states in which  $T$  is true in relation to all 3 doxastic alternatives. We say  $aud$  believes  $T$  with the degree of  $\frac{0}{3}$  what intuitively means the ratio of votes "yes for Alaska" (0) to the overall number of votes (3). In this manner, LGM allows to describe how strong  $aud$  believes in  $T$  - here she is absolutely sure  $T$  is false.

Now, let us focus on how a change is interpreted by studying the meaning of a formula  $\Diamond(i : P)M!_j^{d_1, d_2} T$ . The travel agent, written *prop*, begins his persuasion with giving the argument of his company's special offers on vacation to Alaska ( $a_1$ ). The businesswoman accepts  $a_1$  changing thereby her beliefs - now she considers Alaska option as very interesting. The second box in Figure 2 represents the moment after performing  $a_1$ . The reality moves from  $s_1$  into  $s_2$ . Allowing the option "Alaska" by  $aud$  results in connecting the state  $s_2$  with  $s_8$ . In this way, the agent's belief on the thesis  $T$  is affected - now the woman allows 1 state in which  $T$  is true to all 4 doxastic alternatives ( $M!_{aud}^{1,4} T$ ). This means that after the argument  $a_1$  she is no longer so sure that  $T$  is false (the degree of uncertainty is now  $\frac{1}{4}$ ). The possibility of this **change** is captured by a formula true in the state  $s_1$ , i.e.  $\Diamond(prop : a_1)M!_{aud}^{1,4} T$ . This means that when *prop* starts with giving  $a_1$  in the situation  $s_1$ ,  $aud$  may believe  $T$  in the degree  $\frac{1}{4}$ . Observe that each argument increases her belief in  $T$  - after executing  $a_2$  she believes it with the degree  $\frac{1}{2}$ , after  $a_3$  - with the degree  $\frac{1}{1}$ . Thus, at the end *prop*'s argumentation turns out to be extremely persuasive -  $aud$  becomes absolutely sure that the thesis is true.

Observe that in our approach the persuasion is treated as a process of *adding* and *eliminating* doxastic alternatives and in consequence - the change of beliefs. Moreover, arguments are understood as different actions or tactics which may be used to influence the beliefs. This means that a persuasion may be built on various grounds in our model: deduction, fallacies or even nonverbal actions. The **fallacies** are verbal tricks that are appealing but have little or even nothing in common with the thesis (like the argument

$a_2$  from the example). Generally, a flattery aims rather to make us feel better (that we are creative, original, smart, beautiful) than to give reasons for believing a thesis. In the real-life practice, the persuaders often use the fallacies very efficiently. Similarly, **nonverbal arguments** such as a smile, a hit, a luxuriously furnished office or a visual advertisement (like  $a_3$  from the example) may appear to be very persuasive. A nice picture of a beautiful landscape and animals often says more than thousand words. Notice that defining persuasion not in terms of deduction has its weaknesses and strengths. The disadvantage of such an approach is that some of the elements of the model (especially, concerning possible scenarios of events) must be determined "outside" of the logic itself, i.e., part of information is coded in the interpretation of the program variables what is a part of the agent specification. On the other hand, it seems to be close to the practice of persuasion, where a great deal of effects that this process creates do not depend on its "logic". That is, a valid argumentation is not a guarantee of the success of persuasion. It may fail to succeed, e.g. if an audience is unfamiliar with given formal rules. In this manner, in our model agents are allowed to acquire beliefs similarly to the real-life practice, i.e. on various, even "irrational", grounds moved by a flattery or a visual advertisement.

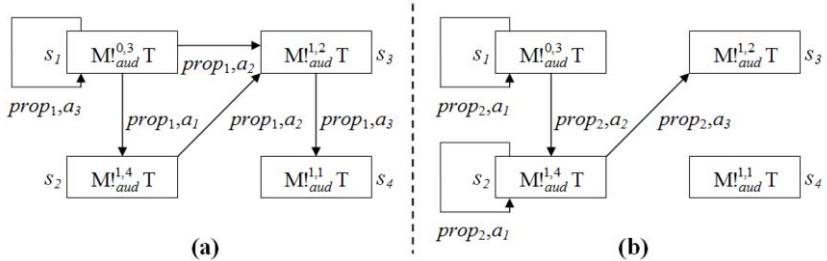
### 3. Interactions amongst agents

In this section, we take subjective aspects of persuasion into account and include them in our formal model. In dialogue systems, the course and the effect of a persuasion depend mainly on used arguments. However, in the real-life practice more important may become who performs these arguments and to whom they are addressed. Treating persuasion as an "impersonal" reasoning can make expressing the changing context of persuasiveness inconvenient or even impossible. Therefore, in our approach the persuasiveness is also related to what agents are engaged in the persuasion: the rank of a proponent and the type of audience. Let us now modify the example from Section 1 in order to illustrate how agents affect the outcomes of persuasion:

*(Example, ct'd) Imagine that the businesswoman is once persuaded by  $prop_1$  working in a well-known travel agency and the other time by  $prop_2$  working in a company about which she heard for the very first time. Both of them use the same sequence of arguments  $a_1, a_2, a_3$ . The woman's reactions on execution of this sequence of actions are different with respect to different executors (i.e. proponents). The "discount" argument  $a_1$  is somehow persuasive when given by  $prop_1$ , but fails when executed by  $prop_2$  since she does not believe that unknown companies can afford good offers. Next given arguments - the "flattery" argument  $a_2$  and the "leaflet" argument  $a_3$  - work for both persuaders.*

The graphs in Figure 3 describe the possible scenarios of the events' course when the businesswoman is persuaded with use of the arguments  $a_1, a_2, a_3$ . The graph (a) models the situations that can happen when persuasion is executed by the travel agent working in the well-known company. The graph (b) shows what will happen in other circumstances, i.e. when the persuader is an agent from the unknown company.

In our model, a graph shows the **interactions amongst agents** rather than only amongst arguments. Its *nodes* (vertices) represent not arguments used to persuade, but states of a system of agents in which audience's beliefs are determined on a basis of a doxastic relation. Further, the *edges* of the graph show not an inference amongst the arguments, but an action which is performed by a proponent with a specific rank. In-



**Figure 3.** The differences in persuasiveness: the same arguments  $a_1, a_2, a_3$ , the same audience  $aud$ , but various proponents - an agent  $prop_1$  in graph (a) and  $prop_2$  in graph (b).

tuitively, a graph can be viewed in our framework as a board of a game which result is determined not only by actions (arguments) we perform, but also what *characters* (a proponent and an audience) we choose to play with. Notice that the nodes and edges of the graphs in Figure 3 carry the information of the names of agents such that these maps stress who takes part in a game. The moves we make along the nodes show how particular arguments given by a specific proponent influence the audience's beliefs.

Now, we are ready to describe the differences in the strength of various persuasions. Recall that persuasion is a process in which a persuader tries to convince an audience to adopt his point of view, i.e., more precisely, to change audience's beliefs. By the strength of a persuasion process we understand the effect of such changes, i.e. how much the considered persuasion can influence the beliefs of agents. It depends on many attributes, however we focus on three of them: the rank of a proponent, the type of an audience as well as the kind and the order of arguments. As a measure of the strength of the persuasion we take the degree of the audience's beliefs concerning a given thesis.

In a multi-agent environment, agents have incomplete and uncertain information and what is more - they can be untrustworthy or untruthful. Thus, it becomes very important which agent gives arguments. Proponent's credibility and reputation affect the evaluation of arguments he gives and as a result they modify the degree of audience's beliefs about the thesis. The exchange of a proponent, while keeping the arguments unaltered, may produce different results. In the language of  $\mathcal{AG}_n$  it is expressed by the formulas  $\Diamond(i : P)\alpha$  and  $\neg\Diamond(j : P)\alpha$  which show that agent  $i$  performing a sequence of arguments  $P$  can achieve a situation in which  $\alpha$  is true while agent  $j$  doing the same has not such a chance. As we said a **rank of a proponent** depends on how much we trust him. In our example, the businesswoman trust  $prop_1$  more than  $prop_2$  regarding the argumentation  $a_1, a_2, a_3$ . Say that we want to study the effects of this argumentation with respect to  $aud$  who at the beginning is absolutely sure that the thesis  $T$  is false. It is illustrated in the graph (a) of Figure 3. We start at the square  $s_1$ . Then, we make moves along the edges that lead from  $s_1$  to  $s_2$  (the result of  $a_1$  performed by  $prop_1$ ), from  $s_2$  to  $s_3$  (when  $prop_1$  performs  $a_2$ ), and lastly from  $s_3$  to  $s_4$  (when  $prop_1$  performs  $a_3$ ). In  $s_4$  the  $aud$  is absolutely sure about  $T$ , i.e.  $M_{aud}^{1,1}T$ . If we choose the proponent  $prop_2$ , the result is not the same since we play on a different game board. The argumentation  $a_1, a_2, a_3$  performed by  $prop_2$  leads us from  $s_1$  to  $s_3$  (see the graph (b) in Figure 3) where  $aud$ 's belief is neutral:  $M_{aud}^{1,2}T$ . This means that  $prop_1$  is much more persuasive than  $prop_2$  with respect to  $aud$  in that specific situation since the same argumentation results for  $prop_1$  in full success while for  $prop_2$  not.

Now, assume that the proponent does not change but he uses different arguments or the same arguments but given in different order. Depending on what arguments are under consideration, the audience decides whether or not and how to change her beliefs. In  $\mathcal{AG}_n$  it is expressed by formulas  $\diamond(i : P_1)\alpha$  and  $\neg\diamond(i : P_2)\alpha$  which say that an agent  $i$  executing a sequence of arguments  $P_1$  achieves  $\alpha$  while performing a sequence  $P_2$  it is not possible. For example, a flattery seems to be more suitable tool for the persuasion of the travel agent than a threat. As we see, the **kind of arguments** strongly affects the strength of a given persuasion. However, even apt arguments may give weaker or stronger success depending on the **order** in which the proponent performs them. Showing at first the leaflet of Alaska could give no effect casting only a suspicion concerning the reasons why  $prop$  does not show leaflets of other places. For instance, in the graph (a) of Figure 3 we have:  $\diamond(prop_1 : a_1; a_2; a_3)M_{aud}^{1,1}T$  but  $\neg\diamond(prop_1 : a_3; a_1; a_2)M_{aud}^{1,1}T$ .

The last factor which influences the strength of the persuasion is the **type of audience**. Consider two scenarios in which proponent and arguments are the same but audiences are different. It may happen that the first audience updates her beliefs while the other does not. Observe that two different game boards should be designed in our model if we want to represent effects that persuasion induces on two distinct audiences. In the example from Section 1, the travel agent is persuasive using the flattery to convince the businesswoman ( $aud_1$ ). However, he could fail when referring the compliment of creativity to a housewife who seeks for popular and proven destinations ( $aud_2$ ). In  $\mathcal{AG}_n$  it may be expressed by formulas  $\diamond(prop : a)B_{aud_1}^{1,1}T$  and  $\neg\diamond(prop : a)B_{aud_2}^{1,1}T$ . To conclude, in our model we capture a fact typical for the real-life practice that only some proponents are able to convince some specific audiences with a specific argument.

#### 4. The logic $\mathcal{AG}_n$

Thus far we have described our model of persuasion. In this section we gather all the elements of  $\mathcal{AG}_n$  logic and give the details of its syntax and semantics.  $\mathcal{AG}_n$  is the multimodal logic of actions and graded beliefs based on elements of Algorithmic Logic (AL) [12], Dynamic Logic (DL) [8] and Logic of Graded Modalities (LGM) [18,7].

Let  $Agt = \{1, \dots, n\}$  be a set of names of *agents*,  $V_0$  be an at most enumerable set of *propositional variables*, and  $\Pi_0$  an at most enumerable set of *program variables*. Further, let ; denote a programme connective which is a sequential composition operator.<sup>4</sup> It enables to compose *schemes of programs* defined as the finite sequences of atomic **actions**:  $a_1; \dots; a_k$ . Intuitively, the program  $a_1; a_2$  for  $a_1, a_2 \in \Pi_0$  means "Do  $a_1$ , then do  $a_2$ ". The set of all schemes of programs we denote by  $\Pi$ . Next components of the language are the modalities. We use modality  $M$  for reasoning about beliefs of individuals and modality  $\diamond$  for reasoning about actions they perform. The intended interpretation of  $M_i^d\alpha$  is that there are more than  $d$  states which are considered by  $i$  and verify  $\alpha$ . A formula  $\diamond(i : P)\alpha$  says that after execution of  $P$  by  $i$  a condition  $\alpha$  may holds.

Now, we can define the set of all **well-formed expressions** of  $\mathcal{AG}_n$ . They are given by the following Backus-Naur form (BNF):

$$\alpha ::= p | \neg\alpha | \alpha \vee \alpha | M_i^d\alpha | \diamond(i : P)\alpha,$$

---

<sup>4</sup>There are considered many program connectives in logics of programs, e.g. nondeterministic choices or iteration operations. However, sequential compositions are sufficient for our needs.

where  $p \in V_0$ ,  $d \in \mathbb{N}$ ,  $P \in \Pi$ ,  $i \in \text{Agt}$ . Other Boolean connectives are defined from  $\neg$  and  $\vee$  in the standard way. We use also the formula  $M!_i^d\alpha$  where  $M!_i^0\alpha \Leftrightarrow \neg M_i^0\alpha$ ,  $M!_i^d\alpha \Leftrightarrow M_i^{d-1}\alpha \wedge \neg M_i^d\alpha$ , if  $d > 0$ . Intuitively it means " $i$  considers exactly  $d$  states satisfying  $\alpha$ ". Moreover, we introduce the formula  $M!_i^{d_1,d_2}\alpha$  which is an abbreviation for  $M!_i^{d_1}\alpha \wedge M!_i^{d_2}(\alpha \vee \neg\alpha)$ . It should be read as " $i$  believes  $\alpha$  with the degree  $\frac{d_1}{d_2}$ ". Thereby, by a **degree of beliefs** of agents we mean the ratio of  $d_1$  to  $d_2$ , i.e. the ratio of the number of states which are considered by an agent  $i$  and verify  $\alpha$  to the number of all states which are considered by this agent. It is easy to observe that  $0 \leq \frac{d_1}{d_2} \leq 1$ .

**Definition 1** Let  $\text{Agt}$  be a finite set of names of agents. By a semantic model we mean a Kripke structure  $\mathcal{M} = (S, RB, I, v)$  where

- $S$  is a non-empty set of states (the universe of the structure),
- $RB$  is a doxastic function which assigns to every agent a binary relation,  
 $RB : \text{Agt} \longrightarrow 2^{S \times S}$ ,
- $I$  is an interpretation of the program variables,  $I : \Pi_0 \longrightarrow (\text{Agt} \longrightarrow 2^{S \times S})$ ,
- $v$  is a valuation function,  $v : S \longrightarrow \{\mathbf{0}, \mathbf{1}\}^{V_0}$ .

Function  $I$  can be extended in a simple way to define interpretation of any program scheme. Let  $I_\Pi : \Pi \longrightarrow (\text{Agt} \longrightarrow 2^{S \times S})$  be a function defined by mutual induction on the structure of  $P \in \Pi$  as follows:  $I_\Pi(a)(i) = I(a)(i)$  for  $a \in \Pi_0$  and  $i \in \text{Agt}$ ,  $I_\Pi(P_1; P_2)(i) = I_\Pi(P_1)(i) \circ I_\Pi(P_2)(i) = \{(s, s') \in S \times S : \exists_{s'' \in S} ((s, s'') \in I_\Pi(P_1)(i) \text{ and } (s'', s') \in I_\Pi(P_2)(i))\}$  for  $P_1, P_2 \in \Pi$  and  $i \in \text{Agt}$ .

The **semantics** of formulas of  $\mathcal{AG}_n$  is defined with respect to a Kripke structure  $\mathcal{M}$ .

**Definition 2** For a given structure  $\mathcal{M} = (S, RB, I, v)$  and a given state  $s \in S$  the Boolean value of the formula  $\alpha$  is denoted by  $\mathcal{M}, s \models \alpha$  and is defined inductively as follows:

$$\begin{aligned} \mathcal{M}, s \models p &\quad \text{iff } v(s)(p) = \mathbf{1}, \text{ for } p \in V_0, \\ \mathcal{M}, s \models \neg\alpha &\quad \text{iff } \mathcal{M}, s \not\models \alpha, \\ \mathcal{M}, s \models \alpha \vee \beta &\quad \text{iff } \mathcal{M}, s \models \alpha \text{ or } \mathcal{M}, s \models \beta, \\ \mathcal{M}, s \models M_i^d\alpha &\quad \text{iff } |\{s' \in S : (s, s') \in RB(i) \text{ and } \mathcal{M}, s' \models \alpha\}| > d, d \in \mathbb{N}, \\ \mathcal{M}, s \models \diamond(i : P)\alpha &\quad \text{iff } \exists_{s' \in S} ((s, s') \in I_\Pi(P)(i) \text{ and } \mathcal{M}, s' \models \alpha). \end{aligned}$$

We say that  $\alpha$  is true in a model  $\mathcal{M}$  at a state  $s$  if  $\mathcal{M}, s \models \alpha$ . In [3] we showed the sound axiomatization of the logic  $\mathcal{AG}_n$  and proved its completeness.

## 5. Investigation of the persuasion systems' properties

In previous sections we have presented the formalism in which many aspects of a persuasion process can be expressed. On its basis we want to examine properties of multi-agent systems in which agents have ability to convince each other. In order to do this we designed and implemented a software system called **Perseus**. It gives an opportunity to study the interactions amongst agents, recognize the factors influencing a persuasion, reconstruct the history of a particular argumentation or determine the effects of verbal and nonverbal arguments.

Now we will briefly describe how Perseus works. Assume that we have a persuasive multi-agent system for which is constructed a model compatible with the one proposed in

the previous sections. Then we analyze this model with respect to given properties. First of all, we build a **multi-graph** which *vertices* correspond to states of a system. Moreover, its *edges* correspond to transitions caused by actions as well as to doxastic relations defined for all agents. As soon as the multi-agent system is transformed into mulit-graph model, Perseus does the research on selected properties expressed in the language of  $\mathcal{AG}_n$ . To this end we introduce the **input question**, i.e. the formula  $\phi$  which is given by the following Backus-Naur form:  $\phi ::= \omega | \neg\phi | \phi \vee \phi | M_i^d \phi | \diamond(i : P) \phi | M_i^? \omega | M_i^{! ?_1 ?_2} \omega$  where  $\omega$  is defined as follows  $\omega ::= p | \neg\omega | \omega \vee \omega | M_i^d \omega | \diamond(i : P) \omega$  and  $p \in V_0$ ,  $d \in \mathbb{N}$ ,  $i \in Agt$ . Unless the  $M_i^? \omega$  and  $M_i^{! ?_1 ?_2} \omega$  components do not appear in the question  $\phi$ , it is a standard expression of the logic being under consideration. In this case the Perseus system simply verifies the thesis  $\phi$  over some model  $\mathcal{M}$  and an initial state  $s$ , i.e. checks if  $\mathcal{M}, s \models \phi$  holds. In other words, Perseus can answer questions like: "Can the persuader convince the audience to the thesis and with what degree?", "Is there a sequence of arguments after performing of which the proponent convinces the audience with degree  $d_1, d_2$  of the thesis?". What is more, Perseus can determine such a sequence and check whether it is optimal in the sense of actions' number, i.e. the minimal and maximal length of such a sequence can be investigated. For example, the input of the Perseus tool could be the question  $\diamond(prop : a_1; a_2; a_3) M_{aud}^{! ?_1 ?_2} T$  which means "What will a degree of the audience's belief on  $T$  be after argumentation  $a_1; a_2; a_3$  performed by  $prop$ ?". In other words we ask the system about values of the symbols  $?_1$  and  $?_2$ , say  $d_1$  and  $d_2$  respectively, such that  $\mathcal{M}, s_1 \models \diamond(prop : a_1; a_2; a_3) M_{aud}^{! d_1, d_2} T$  is satisfied. If we put specific values in the input question instead of question marks then in fact we obtain a verification method which allows us to test multi-agent systems with respect to a given specification.

To conclude, using the Perseus system, a user can study different persuasion processes, compare their strength and find optimal choices of actions. Needles to explain that the specific realization of an argumentative system is rather complex and difficult to perform without the help of a software implementation. The tool like Perseus can be used for **analyzing** such systems and **verifying** their properties. It is especially useful when we focus on the dynamics of persuasion and the change of uncertainty of agents related to a rank of a proponent and a type of an audience.

## Conclusions and future work

In the paper we propose the formal model of multi-agent systems in which persuasion abilities are included as well as the modal logic which can be used to investigate properties of such persuasion systems. Our formal model emphasizes subjective aspects of argumentation what allows us to express how they influence the course and the outcome of this process. Therefore, the persuasiveness is related to the performer of the argumentation and to its addressee. Next, the formal model enables to track the dynamics of persuasion at each stage of the process, i.e. to track the history of how the persuasion modifies the degree of uncertainty of an agent. The degrees of beliefs are represented in terms of Logic of Graded Modalities and their changes are described in terms of Algorithmic and Dynamic Logics. Finally, we allow the process of convincing to be based not only on deduction, but also on fallacies or nonverbal arguments.

The long-range aim of our research is to bridge the gap between the formal theories of persuasion and the approaches focused on practice of persuasion.<sup>5</sup> That is, we plan to systematically loosen the assumptions made in logical models which idealize some of the aspects of that process. This paper provides a key first step towards accomplishing this goal. In particular, it emphasizes the subjective aspects of the persuasion, non-deductive means of convincing or gradation of beliefs important in the real-life practice. In the future, we plan to consequently expand our model especially with respect to the issues: (a) the specification of properties of nonverbal arguments - although they are allowed in our formalism, their characteristics should be better described, (b) the addition of specific axioms describing different aspects of persuasion - in order to make the formal inference stronger in our logic, (c) the other possibilities of formalizing the beliefs' gradation - we plan to compare the expressibility of LGM and the probabilistic logic [6].

## References

- [1] L. Amgoud and C. Cayrol. A model of reasoning based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, (34):197–216, 2002.
- [2] L. Amgoud and H. Prade. Reaching agreement through argumentation: A possibilistic approach. In *9th International Conference on the Principles of Knowledge Representation and Reasoning*, 2004.
- [3] K. Budzyńska and M. Kacprzak. A logic for reasoning about persuasion. In *Proc. of Concurrency, Specification and Programming*, volume 1, pages 75–86, 2007.
- [4] K. Budzyńska and M. Kacprzak. Logical model of graded beliefs for a persuasion theory. *Annales of University of Bucharest. Series in Mathematics and Computer Science*, LVI, 2007.
- [5] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, (77):321–357, 1995.
- [6] R. Fagin and J. Y. Halpern. Reasoning about knowledge and probability. *Journal of the ACM*, 41(2):340–367, 1994.
- [7] M. Fattorosi-Barnaba and F. de Carro. Graded modalities I. *Studia Logica*, 44:197–221, 1985.
- [8] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
- [9] S. Komiak and I. Benbasat. Comparing persuasiveness of different recommendation agents as customer decision support systems in electronic commerce. In *Proc. of the 2004 IFIP International Conference on Decision Support Systems*, 2004.
- [10] S. Kraus, K. Sycara, and A. Enenchik. Reaching agreements through argumentation: a logical model and implementation. *Artificial Intelligence*, 104(1-2):1–69, 1998.
- [11] J.-J. Ch. Meyer and W. van der Hoek. *Epistemic logic for AI and computer science*. Cambridge University Press, 1995.
- [12] G. Mirkowska and A. Salwicki. *Algorithmic Logic*. Polish Scientific Publishers, Warsaw, 1987.
- [13] S. Parsons, C. Sierra, and N.R. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261 – 292, 1998.
- [14] S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *Journal of Logic and Computation*, (13):347–376, 2003.
- [15] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, (15):1009–1040, 2005.
- [16] H. Prakken. Formal systems for persuasion dialogue. *The Knowledge Engineering Review*, 21:163–188, 2006.
- [17] R. Riveret, A. Rotolo, G. Sartor, H. Prakken, and B. Roth. Success chances in argument games: a probabilistic approach to legal disputes. In A.R. Lodder, editor, *Legal Knowledge and Information Systems. JURIX 2007: The Twentieth Annual Conference*. IOS Press, 2007.
- [18] W. van der Hoek. *Modalities for Reasoning about Knowledge and Quantities*. Elinkwijk, Utrecht, 1992.
- [19] D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of N.Y. Press, 1995.

---

<sup>5</sup>See our webpage <http://perseus.ovh.org/> for more details.

# Cohere: Towards Web 2.0 Argumentation

Simon BUCKINGHAM SHUM\*

*Knowledge Media Institute, The Open University, UK*

**Abstract:** Students, researchers and professional analysts lack effective tools to make personal and collective sense of problems while working in distributed teams. Central to this work is the process of sharing—and contesting—interpretations via different forms of argument. How does the “Web 2.0” paradigm challenge us to deliver useful, usable tools for online argumentation? This paper reviews the current state of the art in Web Argumentation, describes key features of the Web 2.0 orientation, and identifies some of the tensions that must be negotiated in bringing these worlds together. It then describes how these design principles are interpreted in *Cohere*, a web tool for social bookmarking, idea-linking, and argument visualization.

**Keywords:** argumentation tools; argument visualization; usability; Web 2.0

## 1. Introduction: The Need for Distributed, Collective Sensemaking Tools

The societal, organizational, scientific and political contexts in which we find ourselves present problems on a global scale which will require negotiation and collaboration across national, cultural and intellectual boundaries. This, I suggest, presents both major challenges and unique opportunities for us, as the community dedicated to understanding computational support for argumentation: our challenge is to work with relevant stakeholders to co-evolve new practices with flexible, usable tools for communities to express how they *agree and disagree* in principled ways, as part of building common ground and mutual understanding.

While our previous work has focused on the real time mapping of issues, dialogue and argument in contexts such as e-science teams [1] and personnel rescue [2], this paper focuses specifically on the challenge of designing engaging, powerful tools for distributed, primarily asynchronous work which, in particular, exploits the strengths of the “Web 2.0 paradigm”. The paper begins by reflecting on the kinds of expectations that Web users and developers now bring, before surveying the current state of the art in Web Argumentation tools. We then describe how Web 2.0 principles, as introduced, have informed the design of a prototype tool called *Cohere*, concluding with a vision of how COMMA researchers might extend or interoperate with it as it moves towards a Web services platform.

---

\* Correspondence to: Knowledge Media Institute, The Open University, Milton Keynes, MK7 6AA, UK.  
Tel: +44 (0) 1908 655723; Fax: +44 (0) 1908 653196; <http://kmi.open.ac.uk/people/sbs>

## **2. The Web 2.0 Paradigm**

A lot is being written about the Web 2.0 paradigm, a term first dubbed in 2004 [3]. While some dismiss it as marketing hype, it does serve as a useful umbrella term to cover significant new patterns of behaviour on the Web. There are many lists of the key characteristics of Web 2.0, not all of which are relevant to our concerns (e.g. e-business models). In this section we select several characteristics for their impact on the user experience of collective information structuring. Together these present a challenge to the design of practical Web Argumentation tools, given the expectations that users now have from their everyday experience of the Web. If we cannot create tools within the new landscape, argumentation tools will remain a much smaller niche than they should be — and as this paper seeks to demonstrate, need be.

### *2.1. Simple but Engaging Multimedia User Interfaces*

The World Wide Web has established itself as the default platform for delivering interactive information systems to professionals and the public. Although early Web applications lacked the elegance and interactivity of desktop applications due to the need for the client to communicate every state change to the server, the gap is closing rapidly with the emergence of good graphic design principles, controlled layout and stylesheet management, and critically, so-called Rich Internet Applications: interactive multimedia capabilities such as Adobe Flash embedded as standard browser plugins, and approaches such as AJAX (Asynchronous JavaScript And XML) for caching local data to increase the responsiveness of the user interface [4]. Users increasingly expect Web applications to have a clean, uncluttered look, and to be as responsive as offline tools. Given a choice of Web offerings, the user experience can determine whether or not a tool is adopted.

### *2.2. Emergent, Not Predefined, Structure and Semantics*

Argumentation focuses on a particular kind of semantic structure for organising elements. Of central interest, therefore, is the Web 2.0 emphasis away from predefined information organizing schemes, towards self-organised, community indexing ('tagging') of elements, resulting in so-called "folksonomies" that can be rendered as tag clouds and other visualizations. Persuading 'normal people' (in contrast to skilled information scientists or ontology engineers) to create structured, sometimes high quality, metadata was previously thought impossible, and the success and limits of this approach is now the subject of a new research field that studies collaborative tagging patterns, e.g. [5].

Another way in which this emphasis expresses itself is in the new generation of tools that make it easy to publish one's opinion of the world. Free, remotely hosted blogging tools such as Wordpress and Blogger make it very easy for non-technical users to create a personally tailored journal or diary and syndicate their ideas. Blogs demonstrate one way to negotiate the formality gulf successfully, providing expressive freedom (essentially, traditional prose and graphics), with just enough structure to reap some benefits of hypertext (entries are addressable as URLs, timestamped, tagged, and syndicated as web feeds – see below). The limitation of blogging at present is that like the Web at large, there are no semantics on the links between postings, thus failing to

provide any support for an analyst who wants to gain an overview of the moves in a debate, or indeed, any kind of inter-post relationship.

### 2.3. Social Networks

Web 2.0 applications are dominated, although not exclusively restricted to, sites that either seek explicitly to connect people with people, often via the artifacts that they share. They are designed such that the greater the numbers participating, the higher the return on effort invested. Social tools provide a range of ways in which users are made aware of peer activity, for instance, alerting when another user ‘touches’ your material (e.g. by reusing it, making it a favourite, tagging it), or by mining social network structure to suggest contacts in a professional network. Social tools also provide mechanisms for building reputation, from the trivial (how many “friends” one has), to potentially more meaningful indices, such as authority based on the quality of material or feedback that a user posts, or professional endorsements.

### 2.4. Data Interoperability, Mashups and Embedded Content

A core idea behind the Web 2.0 paradigm is access to data over the web from multiple applications. Web feeds using RSS and Atom have become the lingua franca for publishing and subscribing to XML data in a simple manner that many non-technical users now handle daily. Public APIs and web services enable the more sophisticated access that enterprise architectures require, while semantic web services promise to overlay ontologies on these layers so that they can be configured according to function. “Mashups” of data sources fuse disparate datasets around common elements (e.g. geolocation, person, date, product), often accessed via customisable user interfaces such as Google Maps [6]. While many mashups typically need to be crafted by a programmer, others can be generated by end-users, given a sufficiently flexible environment. The results of a search may bring together data in new ways.

The phenomenal growth of web applications such as Google Maps, YouTube, Flickr and Slideshare is in part due to the ease with which users can embed remotely hosted material in their own websites. By providing users with the ‘snippet’ code (which may be HTML or JavaScript), such applications empower users to in turn provide their readers with attractively presented access to the material, which can in turn be embedded by those readers in their sites. The material thus spreads ‘virally’, as the links to a resource increase: it is no longer necessary to visit a web page to access its content.

## 3. Web Argumentation Tools

A significant strand in COMMA research focuses on the design, implementation and evaluation of practical software tools for creating and analysing arguments. Following the entity-relationship modelling paradigm that lends itself so well to software, as well as the work of pioneering argument and evidence mapping theorists such as Wigmore and Toulmin, these tools provide a way to construct arguments as structures comprising semantically linked elements taken from one or more argumentation schemes. The argument structures may be left implicit behind text-centric user interfaces, or rendered explicitly as trees or networks to help the author and reader visualize and edit the

argument [7]. The intended users of such tools include members of the public engaged in a public consultations and societal debate [8], students or educators in a learning context [9], lawyers [10], and analysts in many other walks of professional life such as public policy [11] and scholarly publishing [12]. Research in this field examines issues including the translation of argumentation theory into computable representations [13], the nature of expert fluency with such tools [14, 15], and empirical studies of the tools' usage in all of the above domains.

In light of the high design standards and new possibilities that the Web 2.0 paradigm sets, it is clear that existing tools have limitations. First, there are desktop applications like Compendium [30] and Rationale [16] with high quality user interfaces refined through the feedback from their extensive user communities: however, these are limited to publishing read-only maps to the Web, either as JPEG images, or as interactive image maps. Single user applications like CmapTools which have been migrated to 'groupware' versions provide remote editing of maps, but do not exploit the Web 2.0 functions described above.

Finally and most relevant, there are a number of Web-native applications, designed from the start to support large scale, multi-user construction. Some websites now provide a very simple structure for structuring the two sides of a debate, while others provide a more articulated argumentation language. Beginning with the least structured, we see the emergence of sites such as Debatedepedia, which is modelled on Wikipedia, providing a debating resource showing unstructured prose arguments for and against a particular proposal, demarcated in two columns [17]. CoPe\_it! [18] is designed for community deliberation, and provides a way to synchronise views between IBIS graphs (it also integrates with Compendium in this respect), an IBIS outline tree, and a conventional threaded discussion forum. CoPe\_it! also provides a mechanism to evaluate the strength of a position, and so represents another interesting development. Its interaction design is at present rather rudimentary compared to Web 2.0 interfaces. It does not have an end-user customisable semantics, interoperability with existing Web data sources, or mechanisms to syndicate content outside the application.

Parmenides is designed to support web-based policy consultation with the public, and incorporates a formal model of argumentation [19]. It provides a forms-based, questionnaire interface to elicit views from the user, populating an argumentation structure, which it then reasons over to elicit further views. Parmenides enforces a particular argument ontology (it was not designed as a social web application) and does not appear to support any other Web 2.0 characteristics.

ClaiMaker [20] was a Web 1.0 era application, developed in our own prior work modelling the claims and arguments in research literatures. ClaiMaker, and its sister tool ClaimSpotter [21], provided vehicles for us to validate empirically the usability of the data model and a number of user interface paradigms. This has led us to carry the core data model through into Cohere, while relaxing the constraint that restricted users to the predefined classifications of nodes and links. Cohere's visualizations are also versions of those first prototyped in ClaiMaker.

TruthMapping goes much further than this, aiming specifically at tackling some of the limitations of threaded discussion forums, with a clear distinction between unsupported premises, which when supported become claims, and a way to post rebuttals and responses to each of these [22]. DebateMapper uses a combined graphical and outline structure to map debates using the IBIS scheme, with contributions tagged as issues, positions and arguments [23]. DebateMapper perhaps illustrates most clearly some the Web 2.0 interaction design principles, but provides no open semantics, or an

open architecture to enable services on the data. The ArgDF system [24] is the first argumentation tool to adopt a Semantic Web architecture based around the W3C standard Resource Description Framework (RDF) for distributed data modelling and interchange. Moreover, ArgDF is probably the first interactive tool to ground its argument representation in the recently proposed Argument Interchange Format (AIF) [25]. This combination of AIF+RDF is a notable advance. However, while proving the conceptual and technical feasibility of a semantic web orientation for argumentation, it does not yet have a user community, and it cannot be regarded as a Web 2.0 application as defined above.

## 4. The Cohere system

We now describe how we are trying to incorporate the Web 2.0 principles introduced above to create an environment called *Cohere* [[cohereweb.net](http://cohereweb.net)] which aims to be semantically and technically open, provide an engaging user experience and social network, but provide enough structure to support argument analysis and visualization.

### 4.1. Emergent Semantics: Negotiating the Formalization Gulf

In any user-driven content website, the challenge is to keep entry barriers as low as possible to promote the growth of the community, yet maintain coherence of navigation and search, through the careful design of the data model and user interface. The combination of data model and user interface must seek the right balance between constraint and freedom. This Web 2.0 orientation might seem to be in tension with an environment designed to promote rigorous thinking and argumentation. Our approach is to start with relaxed constraints in order to foster engagement with the idea of structuring ideas in general, but provide tools to incrementally add structure as the user recognises the value that it adds in a given context.

Cohere is, therefore, styled to invite playful testing by people who may not first and foremost be interested in argumentation. Instead, the website invites them to *make connections between ideas*. This broader framing aims to meet the need of many sensemaking communities to express how ideas or resources are related (whether or not this is argumentative) in a way that goes beyond plain text blog postings, wikis or discussion forums. A typical pair of connected Ideas in Cohere is illustrated in Figure 1.



Figure 1: Example of a user-defined connection between two Ideas in the Cohere system

In Cohere, users are free to enter any text as an Idea and its detailed description. The examples seeding the database convey implicitly that Idea labels are generally short and succinct. Users are encouraged by the user interface to reuse existing Ideas, with an autocomplete menu dropping down as they type to show matching Ideas already published: as far as possible, we want them to describe the same Idea using the same label.

Users must, however, constrain their contributions by:

- creating labelled connections between Ideas (e.g. *is an example of*)
- reusing, or creating, a connection from a list of either positive, neutral or negative connections

Users can optionally:

- assign roles to Ideas (e.g. *Scenario; Problem*)
- add descriptive details (displayed when the *Info* icon is clicked)
- assign websites to Ideas (listed when the Idea is selected)

The Cohere data model is inherited from the ClaiMaker prototype [11]. The essence is summarised informally in Figure 2.

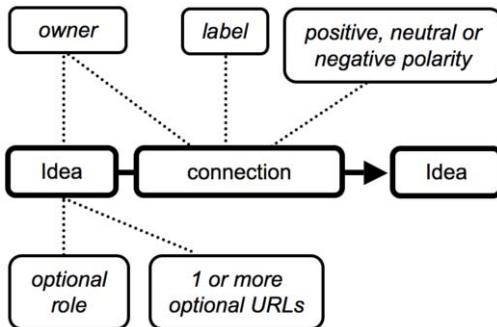


Figure 2: Cohere's data model

The provision of mechanisms to enable flexible linking of web resources around what we are calling Ideas is a goal shared by the Topic Maps community [26], whose data model is very close to that of Cohere. Intriguingly, the two were developed entirely independently, yet arrived at the same core data model, which we take to be a form of empirical validation. In the more formally defined, and more wide-ranging Topic Map Data Model, *topics* (=Cohere Ideas) point to one or more *resources* (=websites); topics can be linked by *associations* (=connections), and topics may play one or more *roles* within a given association (=roles). A Web 2.0 application called Fuzzzy [27] is based on the Topic Map standard and shares some similarities with Cohere, as does the HyperTopic system [28]; neither, however, provide support for argumentation.

While not mandating that the user engage in argumentation, the language of deliberation and argument is nonetheless at the heart of Cohere: (i) the *roles* that Ideas can play in a connection include the IBIS scheme's *Question, Answer, Pro, Con* and the user can define new ones (e.g. *Datum, Claim, Warrant* for Toulmin); (ii) the connection types offered to users are clustered by positive, neutral or negative polarity, with defaults including discourse moves such as *proves, is consistent with, challenges, refutes*. These default connection types are also leveraged in the predefined visualization filters offered by the Connection Net tab, described later (Figure 6). While the interface makes it clear that users may choose to ignore the defaults and create their own connection language, and the roles Ideas can play, the fact that all connections are classed as broadly positive, neutral or negative provides a way to express not only disagreement in the world of discourse, but could signify inhibitory influence (e.g. in biological or ecological systems modelling), or antagonistic

relationships (e.g. in social networks). It is entirely up to the individual or team to define their modelling scheme.

#### *4.2. Visualizing IBIS-Based Dialogue Mapping in Cohere*

The default roles that an Idea can play in a connection are Questions, Answers, Pros and Cons, derived from the Issue-Based Information System (IBIS) developed by Rittel [29], and implemented in the Compendium tool referred to earlier. This is used to model what Walton and Krabbe [30] classified as deliberation dialogues over the pros and cons of possible courses of action to address a dilemma.

Our previous work has demonstrated the value of real time IBIS dialogue mapping in meetings, and the use of IBIS as an organising scheme around which an analyst can map, asynchronously, the structure of public policy debates which can then be published as read-only maps on the Web [31]. Cohere now provides a platform for collaborative deliberation and debate mapping over the internet, with primarily asynchronous use in mind to start with. (Real time mapping requires a tool like Compendium which has a user interface optimised for rapid mapping. However, it is our intention to optimise for real time mapping in the longer term, perhaps by adapting Compendium as an applet for Cohere).

#### *4.3. Visualizing Argumentation Schemes and Critical Questions in Cohere*

In related work [32], we have demonstrated how Walton's argumentation schemes and associated Critical Questions, rendered as XML files in the Argument Markup Language [33], can be transformed into Compendium XML and expressed as IBIS structures in Compendium. The resulting argumentation scheme templates can now be modelled in Cohere as illustrated in Figure 3.

#### *4.4. Social Networking and Reputation*

All Ideas other than one's own have their owner clearly indicated iconically. Clicking this displays the user profile, making it possible to learn more about the person behind the ideas. We are beginning to add metrics to make users aware when they arrive at the site how many active users there are, and what the most recently posted, reused and linked Ideas are. Web feeds in the future will enable users to request notification whenever one of their Ideas is embedded in someone else's connection, or in someone else's website (see below).

#### *4.5. Interoperability: Web Data as Platform*

Central to O'Reilly's notion of Web 2.0 is the notion of web data as the platform on which many applications can compute. Cohere exposes and consumes data in a variety of ways:

- Publishing and importing XML Web feeds
- Importing XML data from the Compendium offline dialogue and argument mapping tool
- Embedding pointers to its data in other applications as URLs and HTML 'snippets'
- Exposing data in a variety of standards to engage different communities

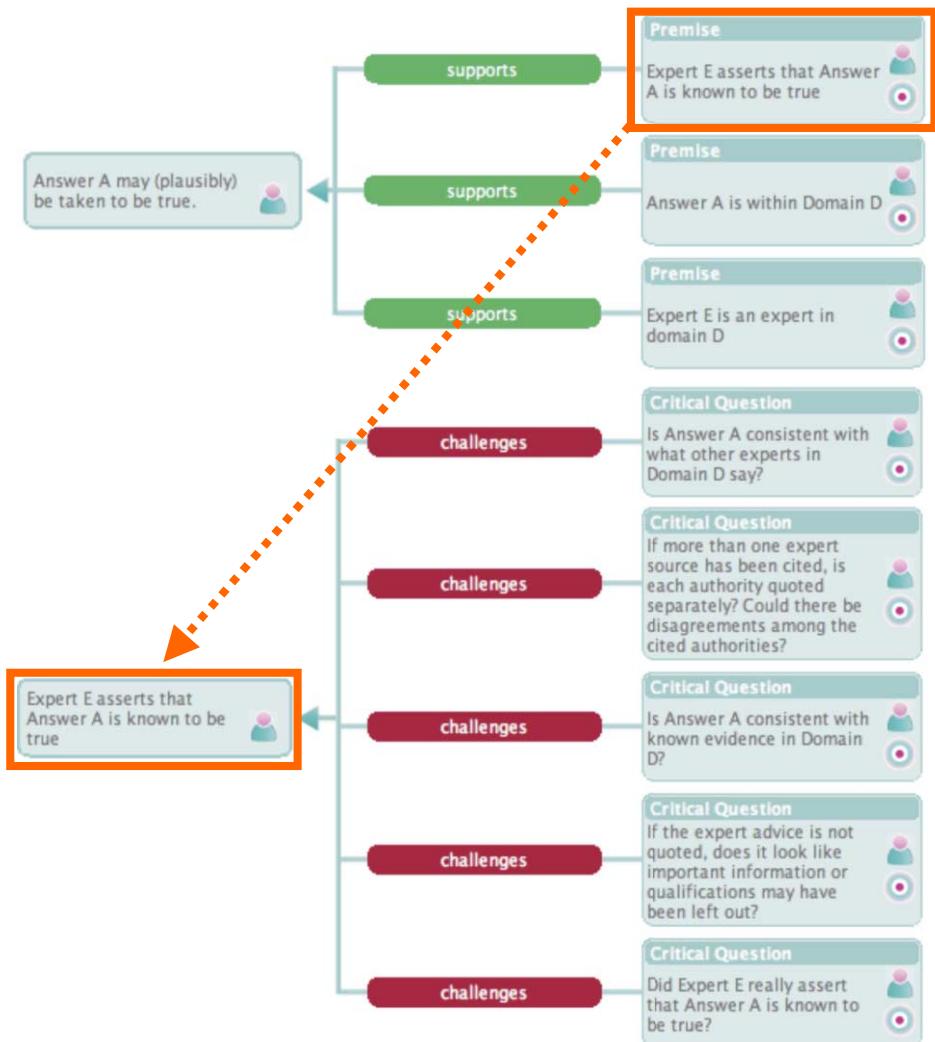


Figure 3: Rendering Walton's Critical Questions on the *Argument from Expert Opinion* scheme, as an IBIS

**Web feeds:** Cohere seeks to build on the significant effort that many users already invest in social bookmarking with folksonomic tagging tools such as del.icio.us, or in blogging with tools such as Blogger or Wordpress. These are currently two of the most dominant ways in which users share their views, and Cohere aims to leverage this by enabling users to import/refresh the Web feed (RSS or Atom) for any bookmarking or blogging site. Entries are converted into *Ideas* and annotated with the relevant URL, ready for linking. We are in the process of implementing an RSS feed so that users can track new Ideas as they are published. We plan to develop this capability, so that individual Ideas can be subscribed to, with alerts everytime someone connects to or from them.

**Ideas and views as URLs:** It is increasingly hard to find an artifact or building these days without a URL on it. The web depends on the URL as a way for non-

technical users to connect web pages, save searches, and disseminate sites of interest via standard tools such as email, slides and wordprocessors. The design of URLs goes beyond cool top level domain names, to the art of URLs that communicate their content to people, in contrast to machine-generated addresses that have no obvious pattern.

It was considered essential, therefore, to make Cohere's content addressable and accessible as URLs. This required the creation of a guest login status for non-registered users to successfully reach an address, and the design of a URL syntax that specified the visualization type and search criteria. The URL for an Idea, a triple, or a Connection List/Net is accessed by the user in what has become the conventional manner, by clicking on a URL icon to copy and paste the address that pops up.

**Embedding ideas and views in other websites:** Once a URL addressing scheme is in place, it becomes possible to provide such embeddable snippets for users, as introduced above. Pasting this <iframe> code into a web page creates an embedded, interactive view onto the Cohere database, which reproduces the buttons to get the URL and snippet code, to encourage further dissemination (Figure 4).

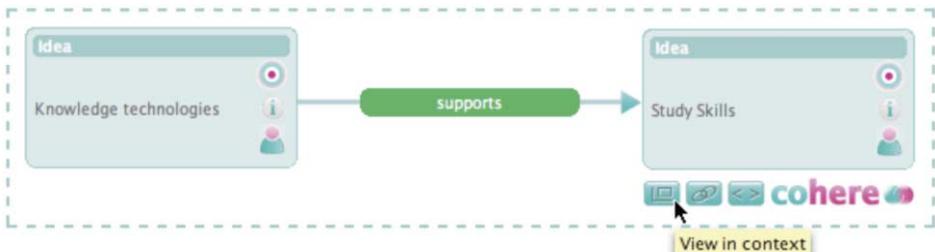


Figure 4: A Cohere connection embedded as a snippet in another web page. The three buttons below take the user to the connection within the Cohere website, provide the URL to this link, and provide the HTML embed code. Users can embed single Ideas, or whole interactive maps.

**Multiple Import/Export data formats:** By the time of the conference, we will have implemented further data formats for importing and exporting Cohere structures. A priority is to provide Argument Interchange Format compatibility, with other candidates being Topic Maps, Conceptual Graphs, and OWL.

#### 4.6. Mashup Visualizations

Our objective is to help forge links not only between Ideas, but between the people publishing them. As Cohere starts to be used, it is inevitable that popular Ideas will be duplicated: if the site is successful, we can expect many people to be working on the Idea *Global Warming*, or making reference to everyday concepts such as *Capitalism* or *World Wide Web*. We have started to design views that help render the structures that will result from many users working on common Ideas. This is a long term challenge, but Figure 5 shows the first tool called Connection Net, which uses a self-organising graph layout algorithm that can render all of one's personal data, or filtered views of the world's data. In particular, Ideas with a border are used by more than one person, and as shown, right-clicking on it enables the user to view all the owners of that Idea. In this way, just as folksonomies enable disparate users to discover related resources and people, Cohere aims to reveal new connections and users working on the same Idea, or perhaps more interestingly, making similar or contrasting connections.

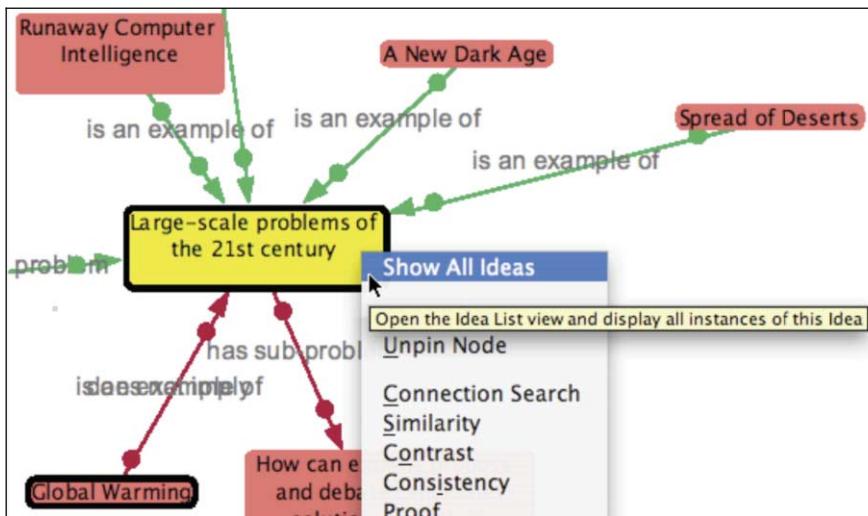


Figure 5: The Connection Net view merges all matching Ideas in a single node, and lays out the graph automatically

Filter buttons in the Connection Net view make use of the connection types, as shown in Figure 6. A number of saved filters are shown, for example, *Contrast* searches the database from a focal Idea on a specific subset of connection types of a contrasting nature, e.g. *challenges*, *has counterexample*, *is inconsistent with*, *refutes*. Users can define their own custom searches, and in the future will be able to save them as shown in the example buttons.

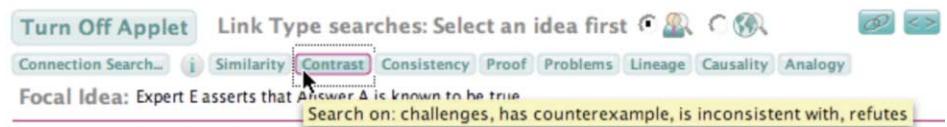


Figure 6: Semantic filter buttons that show only a subset of connection types from a focal Idea. The example shown is a Contrast search: rolling over it displays the connection semantics it will search on. User defined searches are issued from the Connection Search button on the left.

#### 4.7. Implementation

Cohere is implemented on Linux, Apache HTTP server, MySQL database, and PHP. The user interface exploits the AJAX approach to caching data in the browser to create a highly responsive interface, with few delays between action and feedback. Cascading Style Sheets are used extensively to control presentation. In addition, a Java applet from the Prefuse information visualization classes [34] has been customised to provide self-organising, interactive graph visualizations under the Connection Net tab. Compendium (op cit) serves as an offline mapping tool (a cross-platform Java desktop application with Apache Derby or MySQL database). Data is uploaded to Cohere currently using the Compendium XML scheme. Cohere is currently a freely hosted application, and an open source release is planned by end of 2008.

## 5. Present Limitations, and Future Work

This project is tackling a complex but important challenge: to create tools providing a compelling user experience by harnessing two forces that seem on first inspection to pull in opposite directions: on the one hand, informal social media with low entry thresholds and few interaction constraints, and on the other, mechanisms for structuring ideas and discourse. We have presented the design rationale behind Cohere, a web application for structuring and visualizing information and arguments, publishing Ideas, and discovering new intellectual peers. In order to balance the informal+formal design criteria, we bias to the informal, with interface and architectural mechanisms to add structure as desired. Understandably, Cohere is being trialled initially by individuals testing it for personal reflection and information management, but ultimately, we hope to augment distributed communities engaged in intentional, collective sensemaking.

We are now in discussion with other COMMA research groups to explore how their argument modelling approaches could be integrated with Cohere. We are moving to a web services architecture [cf. 35] and plan to enable data exchange via the W3C's RDF and OWL, and the proposed Argument Interchange Format [25]. These developments are designed to evolve Cohere from a closed web application, towards a collaboration platform for structured, social argumentation and deliberation for wider experimentation by both end-users, developers and argumentation researchers. Future technical work will support different argument layouts, more flexible visualizations, group permissions, and the management of template 'pattern libraries' (currently managed via the offline Compendium tool). Our pilot usability evaluations are leading to interface changes that are being added at the time of writing, and will be followed by more in depth studies in the lab (cf. [12]), and with end-user communities.

Finally, our goal with Cohere is to provide an environment for the emergence of social and argument structures. While not currently exposed in the user interface, Cohere has built into its data model the Cognitive Coherence Relations modelling scheme described in the COMMA'08 debate modelling paper by Benn *et al.* [36], which seeks an integrated approach to modelling social networks and argument networks. A future objective is to investigate this modelling paradigm within Cohere.

**Acknowledgements:** Cohere is implemented by Michelle Bachler, with contributions from Michele Pasin and Alex Little, to whom I am indebted. The *Open Sensemaking Communities* Project [[www.kmi.open.ac.uk/projects/osc](http://www.kmi.open.ac.uk/projects/osc)] is funded by the William and Flora Hewlett Foundation, as part of the Open University's *OpenLearn* Initiative [[www.open.ac.uk/openlearn](http://www.open.ac.uk/openlearn)].

## 6. References

- [1] Sierhuis, M. and Buckingham Shum, S. (2008). Human-Agent Knowledge Cartography for e-Science. In: *Knowledge Cartography*, (Eds.) Okada, A., Buckingham Shum, S. and Sherborne, T. Springer
- [2] Tate, A., Buckingham Shum, S., Dalton, J., Mancini, C. and Selvin, A. (2006). Co-OPR: Design and Evaluation of Collaborative Sensemaking and Planning Tools for Personnel Recovery. *Technical Report KMI-06-07*, Open University, UK. [www.kmi.open.ac.uk/publications/pdf/KMI-TR-06-07.pdf](http://www.kmi.open.ac.uk/publications/pdf/KMI-TR-06-07.pdf)
- [3] O'Reilly, T. and Battelle, J. (2004). The State of the Internet Industry. Opening Presentation, *Web 2.0 Conference*, Oct. 5-7 2004, San Francisco, CA.
- [4] Garrett, J.J. (2005). Ajax: A New Approach to Web Applications. *Adaptive Path* (Feb. 18, 2005): [www.adaptivepath.com/ideas/essays/archives/000385.php](http://www.adaptivepath.com/ideas/essays/archives/000385.php)
- [5] Golder, S. and Huberman, B.A. (2006). Usage Patterns of Collaborative Tagging Systems. *Journal of Information Science*, 32(2). 198-208.

- [6] See for instance the *Crisis in Darfur* project by the The United States Holocaust Memorial Museum and Google Earth: [www.ushmm.org/googleearth](http://www.ushmm.org/googleearth)
- [7] On argument visualization see for instance the volume in [15] and the recent conference *Graphic and Visual Representations of Evidence and Inference in Legal Settings*: [www.tillers.net/conference.html](http://www.tillers.net/conference.html)
- [8] *Argumentation Support Systems for eParticipation*. EU DEMO-net Project: [www.demo-net.org](http://www.demo-net.org)
- [9] Andriessen, J., Baker, M. and Suthers, D. (2003). *Arguing to Learn*. Kluwer
- [10] Verheij, B. (2005). *Virtual Arguments. On the Design of Argument Assistants for Lawyers and Other Arguers*. T.M.C. Asser Press, The Hague
- [11] Lowrance, J. (2007) Graphical Manipulation of Evidence in Structured Arguments. *Oxford Journal of Law, Probability and Risk*
- [12] Uren, V., Buckingham Shum, S.J., Li, G. and Bachler, M. (2006) Sensemaking Tools for Understanding Research Literatures: Design, Implementation and User Evaluation. *Int. Jnl. Human Computer Studies*, **64**, (5), 420-445.
- [13] Reed, C., Walton, D. & Macagno, F. (2007). Argument diagramming in logic, law and artificial intelligence. *Knowledge Engineering Review*, **22** (1), 87-109.
- [14] Conklin, J. (2006). *Dialogue Mapping*. Chichester: Wiley
- [15] Selvin, A. (2003). Fostering Collective Intelligence: Helping Groups Use Visualized Argumentation. In *Visualizing Argumentation*, (Eds.) P. Kirschner, S. Buckingham Shum, and C. Carr. Springer: London
- [16] van Gelder, T. (2007). The Rationale for RationaleTM. *Oxford Journal of Law, Probability and Risk*.
- [17] *Debatepedia*: International Debate Education Association: [wiki.idebate.org](http://wiki.idebate.org)
- [18] Karacapilidis, N. and Tzagarakis, M. (2007). Web-based collaboration and decision making support: A multi-disciplinary approach. *Int. J. Web-Based Learning and Teaching Technologies*, **2**, (4), 12-23.
- [19] Atkinson, K., Bench-Capon, T. and McBurney, P. (2006): PARMENIDES: Facilitating deliberation in democracies. In: *Artificial Intelligence and Law*, **14** (4), pp. 261-275.
- [20] Buckingham Shum, S.J., Uren, V., Li, G., Sereno, B. and Mancini, C. (2007). Modelling Naturalistic Argumentation in Research Literatures: Representation and Interaction Design Issues. *International Journal of Intelligent Systems*, **22**, (1), pp.17-47.
- [21] Sereno, B., Buckingham Shum, S. and Motta, E. (2007). Formalization, User Strategy and Interaction Design: Users' Behaviour with Discourse Tagging Semantics. Workshop on Social and Collaborative Construction of Structured Knowledge, *16th Int. World Wide Web Conference*, Banff, AB, Canada
- [22] *TruthMapping*: [www.truthmapping.com](http://www.truthmapping.com)
- [23] *DebateMapper*: [www.debatemapper.com](http://www.debatemapper.com)
- [24] Rahwan, I., Zablith, F. and Reed, C. (2007). Laying the Foundations for a World Wide Argument Web. *Artificial Intelligence*, **171**, (10-15), pp 897-921
- [25] Chesnevar, C., McGinnis, J., Modgil, S., Rahwan, I., Reed, C., Simari, G., South, M., Vreeswijk, G., Willmott, S. (2006) Towards an Argument Interchange Format, *Knowledge Engineering Review*, **21** (4), 293-316.
- [26] ISO 13250-2: Topic Maps — Data Model: [www.isotopicmaps.org/sam](http://www.isotopicmaps.org/sam)
- [27] Lachica, R. and Karabeg, D. (2007). Towards holistic knowledge creation and interchange Part I: Socio-semantic collaborative tagging. *Proc. Third International Conference on Topic Maps Research and Application*, Leipzig. Lecture Notes in Artificial Intelligence, Springer: Berlin
- [28] Zaher, L., Cahier, J.-P., and Guittard, C. (2007) Cooperative Building of Multi-Points of view Topic Maps with Hypertopic. *Proc. Third Int. Conf. Topic Maps Research and Application*, Leipzig. Springer
- [29] Rittel, H.W.J., Second Generation Design Methods. In: *Developments in Design Methodology*, N. Cross (Ed.), 1984, pp. 317-327, J. Wiley & Sons: Chichester
- [30] Walton, D.N. and Krabbe, E.C.W. (1995). *Commitment in Dialogue*, SUNY Press.
- [31] Ohl, R. (2008). Argument Visualisation Modelling in Consultative Democracy around Wicked Problems In: *Knowledge Cartography*, (Eds.) Okada, A., Buckingham Shum, S. and Sherborne, T. Springer: London
- [32] Buckingham Shum, S. and Okada, A. (2008, in press). Knowledge Cartography for Controversies: The Iraq Debate. In: *Knowledge Cartography*, (Eds.) Okada, A., Buckingham Shum, S. and Sherborne, T. Springer: London
- [33] Walton, D.N. & Reed, C.A. (2002) Diagramming, Argumentation Schemes and Critical Questions. *Proc. 5th International Conference on Argumentation (ISSA'2002)*, SicSat, Amsterdam, pp. 881-885.
- [34] *Prefuse*: interactive information visualization toolkit: [www.prefuse.org](http://www.prefuse.org)
- [35] EU ASPIC Project: *Argument Service Platform with Integrated Components*: [www.argumentation.org](http://www.argumentation.org)
- [36] Benn, N., Buckingham Shum, S., Domingue, J. and Mancini, C. (2008). Ontological Foundations for Scholarly Debate Mapping Technology. *Proc. 2nd Int. Conf. on Computational Models of Argument*. Toulouse, May 2008. IOS Press: Amsterdam.

# On the Issue of Contraposition of Defeasible Rules

Martin CAMINADA<sup>a</sup>

<sup>a</sup> University of Luxembourg

**Abstract.** The past ten years have shown a great variety of approaches for formal argumentation. An interesting question is to which extent these various formalisms correspond to the different application domains. That is, does the appropriate argumentation formalism depend on the particular domain of application, or does “one size fits all”. In this paper, we study this question from the perspective of one relatively simple design consideration: should or should there not be contraposition of (or modus tollens) on defeasible rules. We aim to show that the answer depends on whether one is considering *epistemical* or *constitutive* reasoning, and that hence different domains require fundamentally different forms of defeasible reasoning.

**Keywords.** epistemical reasoning, constitutive reasoning, contraposition

## 1. Introduction

Recent years have brought a large variety of research regarding argumentation formalisms. These formalisms tend to differ in the way that arguments are constructed and the defeat relation is defined [5,20,6], as well as in the abstract argumentation semantics [9,1,8], and in the various additional forms of functionality that can be provided [4,15].

Given the wide variety of formalisms and approaches, one can ask the question how these relate to each other as well as the question of how this variety is related to the different domains in which argumentation can be applied. As for the first question, some work has already been done by for instance Baroni and Giacomin, who provide a number of abstract properties according to which various argumentation semantics can be compared [3,2]. As for the second question, much less work has been done. Moreover, there is little clarity on whether one should aim for a general argumentation formalism that is applicable to a large variety of domains, or whether different domains require different and perhaps fundamentally incompatible classes of reasoning.

In this paper, it will be argued that the approach of “one size fits all” has serious disadvantages. We claim that there are two fundamentally different forms of reasoning, *epistemical* and *constitutive*, and that these require fundamentally different properties regarding formal entailment.

## 2. Default Contraposition in Epistemical Reasoning

Although the issue of applicability of default contraposition seems to be a fundamental one, it has until now received relatively little attention. Many authors treat the validity or

invalidity of contraposition as an aside when describing their respective formalisms for non-monotonic reasoning. Our analysis will therefore begin with an overview of some of the comments by various authors in the field. Brewka, for instance, provides the following counterexample against the validity of contraposition: “Men usually do not have beards, but this does not mean that if someone does have a beard, it’s usually not a man.” Another example would be: “If I buy a lottery ticket then I will normally not win any price, but this does not mean that if I *do* win a price, I did not buy a ticket.”

Given the last two examples, it seems that there are perfectly legitimate situations in which contraposition does not hold, and that contraposition (or modus tollens) should therefore be rejected as a general principle for defeasible reasoning. The point is, however, that once one starts to accept counterexamples against default contraposition, then one should also take into consideration counterexamples against various other principles for defeasible reasoning:

**irrelevance** “Tux the bird”: Birds fly and Tuxes are birds.<sup>1</sup> Do Tuxes fly? Perhaps not, because Tuxes may belong to a special subclass of birds that do not fly.

**left conjunction** “jogging in the rain” [19]: If it is hot, I tend not to go out jogging. If it is raining I also tend not to go out jogging. Does this mean that if it is hot *and* it is raining, I tend not to go out jogging?

“Marry both of them” [17] If you marry Ann you will be happy, if you marry Nancy you will be happy as well. Does this mean you will be happy if you marry both of them?

**transitivity** “unemployed students” [16] Students are usually adults and adults are usually employed. Does this mean that students are usually employed?

The above counterexamples against irrelevance, left conjunction, contraposition and transitivity look appealing at first sight. The point of each counterexample, however, is that it involves implicit background information. Tux does not fly because it is a penguin; marrying two persons generally does not make one happy (one may end up in jail instead); women have no beards at all; and students are a special class of adults that tend to be unemployed. The view that the counterexamples against contraposition, like the ones above, are flawed is shared by Ginsberg [10, p. 16], although he treats a different example himself (“Humans usually do not have diabetics”).<sup>2</sup>

The “all or nothing” approach to the above mentioned properties of irrelevance, left conjunction, transitivity and contraposition is confirmed when one tries to give the defeasible rules a statistical interpretation. In  $\varepsilon$ -semantics [17], for example, none of these principles are satisfied, whereas the Maximal Entropy approach [11] satisfies all of them, but as defeasible principles only (that is, their instances can be blocked if specific information against their applicability is available). It appears that if one wants to make a consistent choice that includes the (defeasible) validity of properties like irrelevance, left conjunction and transitivity, then one should accept the (defeasible) validity of contraposition as well. Yet, it is striking to see that formalisms for defeasible reasoning tend not to be based on any consistent choice on these issues. A similar observation can be made

---

<sup>1</sup>Tux is the well-known penguin logo of the Linux-community.

<sup>2</sup>As an aside, it appears that the “counterexamples” against contraposition involve rules where the antecedent contributes negatively to the consequent. That is, the consequent holds in spite of the antecedent. See [7] for a more elaborate discussion.

regarding the to contraposition related principle of *modus tollens*. Both Reiter's default logic [22] and the formalism of Prakken and Sartor [20] sanction a defeasible form of modus ponens, but do not sanction any form of modus tollens. A systematic analysis of the actual meaning of a default is often not provided. Yet, it is this analysis that should serve as a basis for determining which principles should or should not be sanctioned. The current trend seems to be to sanction various principles, but not those of (defeasible) modus tollens or contraposition. It is an anomaly that is rarely questioned, and one may wonder whether this is because many researchers have become acquainted with it. Or, as Ginsberg states when discussing the reasons behind the opposition against contraposition [10, p. 16]:

(...) although almost all of the symbolic approaches to nonmonotonic reasoning do allow for the strengthening of the antecedents of default rules, many of them do *not* sanction contraposition of these rules. The intuitions of individual researchers tend to match the properties of the formal methods with which they are affiliated.

### 3. Default Contraposition in Constitutive Reasoning

In the current section, we again ask the question whether contraposition should be sanctioned, this time not from the perspective of probabilistic empirical reasoning, but from the perspective of *constitutive* reasoning. The difference between these two forms of reasoning can perhaps best be illustrated using a mirror example, which is a small logical formalization that can be given two informal interpretations with opposite conclusions [7, section 2.2.5].

*Informal Interpretation 1 (II<sub>1</sub>):* The goods have been ordered three months ago (*TMA*) and the associated customs declaration is still lacking in the customs information system (*LIS*). If the goods have been ordered three months ago, then they will probably have arrived by now ( $TMA \Rightarrow A$ ). If the goods have arrived, then there should be a customs declaration for them ( $A \Rightarrow CD$ ). If the registration of the customs declaration is still lacking in the customs information system, then there probably is no customs declaration ( $LIS \Rightarrow \neg CD$ ).

*Informal Interpretation 2 (II<sub>2</sub>):* John is a professor (*P*) who is snoring in the university library (*S*). Snoring in public is usually a form of misbehaviour ( $S \Rightarrow M$ ). People who misbehave in the university library can be removed ( $M \Rightarrow R$ ). Professors cannot be removed ( $P \Rightarrow \neg R$ ).

In the “arrival of goods” example (*II<sub>1</sub>*) it seems reasonable to apply contraposition on  $A \Rightarrow CD$  to construct an argument for  $\neg A$ . In the “snoring professor” example (*II<sub>2</sub>*), however, it would be very strange to have contraposition on  $M \Rightarrow R$  since this would allow us to construct an argument for  $\neg M$ . In fact, example *II<sub>2</sub>* has been taken from [18, p. 185] where it is claimed that the justified conclusions should include *M*, but not *R* or  $\neg R$ . Hence, the above pair (*II<sub>1</sub>*, *II<sub>2</sub>*) can be considered as a mirror example in the sense of [7, section 2.2.5]. The next question then is how this situation should be dealt with. That is, do we (1) reject at least one of the formalizations as “incorrect” or (2) acknowledge that the two examples are related to fundamentally different forms of reasoning? In this paper, we choose for the second option. That is, we claim that there is a fundamental difference that makes contraposition applicable to *II<sub>1</sub>* but not to *II<sub>2</sub>*.

### *Direction of fit*

In order to understand the nature of constitutive reasoning, it is useful to distinguish between statements that have a *word to world* direction of fit, and statements that have a *world to word* direction of fit [24,25]. It should be noted that also one of the differences between  $II_1$  and  $II_2$  concerns the direction of fit.

The defeasible rules of  $II_1$  are meant to describe when a certain fact holds in the object-world. This object-world has an existence that is independent of the rules that express our knowledge about it. These rules, therefore, have a *word to world* direction of fit. Their correctness depends on a validity that has an independent existence.

In  $II_2$ , on the other hand, the very nature of the rules is different. The rules do not merely describe the reality, but to some extent also construct it, especially if we assume these rules to be taken from, say, the library regulations. The rule  $S \Rightarrow M$ , for instance, contributes to the definition of misbehavior in the context of the library regulations. The rule essentially *makes* it the case that snoring is considered to be misbehavior, as far as the library is concerned. The defeasible rules of  $II_2$ , therefore, have a *world to word* direction of fit. Their application results in the creation of new (legal) facts.

### *Epistemic versus constitutive reasoning*

Based on the direction of fit, one can distinguish two kinds of reasoning: epistemic and constitutive<sup>3</sup>. The nature of this distinction can be described as follows [12, p. 60]: “Epistemic reasons are reasons for believing in facts that obtain independent of the reasons that plead for or against believing them. Constitutive reasons, on the contrary, influence the very existence of their conclusions”.

In order to understand the differences between epistemic and constitutive reasoning, we provide the following abstract example<sup>4</sup> (AE):  $\mathcal{P}remises = \{A; D\}$ ,  $\mathcal{D}efeasible\ rules = \{A \Rightarrow B; B \Rightarrow C; D \Rightarrow \neg C\}$  conflict:  $A; A \Rightarrow B; B \Rightarrow C$  versus  $D; D \Rightarrow \neg C$

Now, take the following two constitutive interpretations of this example.

**deontic** The following example is somewhat similar to that of the Christian Soldier. An artillery soldier is given the order to destroy an enemy military installation, and orders should generally be obeyed ( $order \Rightarrow O(shoot)$ ). When the soldier looks through his binoculars, he observes some movements that probably mean that some people are really close to the target ( $movements \Rightarrow people$ ), thus making it from an ethical point of view imperative not to shoot ( $people \Rightarrow O(\neg shoot)$ ). Thus, we have:  $\mathcal{P}remises : \{order, movements\}$  and  $\mathcal{D}efeasible\ rules = \{movements \Rightarrow people; people \Rightarrow O(\neg shoot); order \Rightarrow O(shoot)\}$ . Conflict:  $movements; movements \Rightarrow people; people \Rightarrow O(\neg shoot)$  versus  $order; order \Rightarrow O(shoot)$

Here, the conflict is between the obligation to shoot and the obligation not to do so. In some logics, like Standard Deontic Logic, such a conflict would lead to an inconsistency. If we would allow for contraposition, the effect would be that  $people$

<sup>3</sup>The term “constitutive rules” was originally introduced by Searle [23]. In this essay, however, we use the term in the sense of [13,12].

<sup>4</sup>The reader will notice that the structure of this example is similar to  $II_1$  and  $II_2$ .

is no longer justified. This is, of course, absurd; the belief in empirical statements should not depend on the presence or absence of deontic conflicts.

**legal** An example of a legal interpretation is  $II_2$ . Here, the reasoning concerns whether or not certain legal facts obtain. Even though the conflict could be described in deontic terms (is the library personnel permitted to remove the person in question or not), the conflict ( $\text{Permitted}(\text{remove})$  v.s.  $\neg\text{Permitted}(\text{remove})$ ) is essentially not of a deontic nature, like in the previous example ( $\text{Obligated}(\text{shoot})$  v.s.  $\text{Obligated}(\neg\text{shoot})$ ). The question is whether it is legally permitted to remove the person or not, and this question does not rely on the specifics of deontic reasoning. The fact that this conflict exists, however, is no reason to reject the intermediate conclusion of  $M$ . To make this point more clear, suppose that the library regulations contain an additional rule saying that those who misbehave have to pay a fine of ten euro ( $M \Rightarrow F$ ) and that no rule is available that provides professors with exemption for this fine. Then, the fact that the intermediate conclusion  $M$  can lead to  $R$  (which conflicts with  $\neg R$ ) is no reason to disallow the entailment of  $F$ .

The point is that constitutive reasoning obeys different principles than epistemic reasoning. Under epistemic reasoning it is perfectly reasonable to sanction contraposition, as was argued in section 2. Under constitutive reasoning, on the other hand, contraposition is *not* valid by default, as was discussed above. In legal reasoning, for instance, the leading paradigm is that the law should be interpreted as consistently as possible. Hence, in the snoring professor example the potential conflict between  $R$  and  $\neg R$  is not a reason to create a conflict between  $M$  and  $\neg M$  and hence reject  $M$  or  $F$ . The idea is to keep the effects of possible conflicts as local as possible [12, p. 109]. A great deal of research has been dedicated at stating and formalizing meta-principles (such as *lex posterior*, *lex specialis* or *lex superior*) for determining which of the conflicting rules should be applied, and which should not. But even in the case that no determining meta-principle is available, the application of *both* rules is blocked and the conflict does not have consequences for conclusions that do not depend on it. Our snoring professor, even though he may not be removed, still has to pay his 10 euro fine.

### (Im)perfect procedures versus pure procedures

The difference between epistemic and constitutive reasoning is comparable to the difference between (im)perfect procedures and pure procedures, as distinguished by Rawls. To illustrate the concept of a *perfect procedure*, Rawls provides the example of cake-cutting [21, p. 74]:

A number of men are to divide a cake: assuming that the fair division is an equal one, which procedure, if any, will give this outcome? Technicalities aside, the obvious solution is to have one man divide the cake and get the last piece, the others being allowed their pick before him. He will divide the cake equally, since in this way he assures for him the largest share possible. This example illustrates the two characteristic features of perfect procedural justice. First, there is an independent criterion for what is a fair division, a criterion defined separately from and prior to the procedure which is to be followed. And second, it is possible to devise a procedure that is sure to give the desired outcome.

One of the assumptions of the above cake-cutting example is that the person cutting the cake can do so with great accuracy. As long as deviations in cutting are ignored, the result will be an equal distribution. If we assume that the deviations in cutting cannot be ignored, cake-cutting becomes an *imperfect procedure*. The characteristic mark of an imperfect procedure is that while there is an independent criterion for the correct outcome, there is no feasible procedure which is sure to lead to it [21, p. 75].

A *pure procedure*, on the contrary, is the case when there is no independent criterion for the right result: instead there is a correct or fair procedure such that the outcome is likewise correct or fair, whatever it is, provided that the procedure has been properly followed. An example of a pure procedure is a free election. The outcome of elections cannot be evaluated as “right” or “wrong” according to an outside objective standard. The idea is that any resulting outcome should be accepted, as long as the election process itself was carried out in a correct way. In general, one can only fight the outcome of a pure procedure by arguing that the procedure itself was not applied properly [14].

Since the process of reasoning can to some extent be seen as a procedure, it is interesting to evaluate how the kind of reasoning as performed in  $II_1$  and  $II_2$  can be seen in terms of (im)perfect and pure procedures.

$II_1$  is basically an instance of empirical (epistemic) reasoning. One uses potentially incomplete information and rules of thumb, with the idea that the reasoning process is likely to generate a correct result. Even though an outside criterion exists to evaluate correctness (the goods have either arrived or not), there is no guarantee that the reasoning process indeed obtains this result. Hence, the reasoning process as performed in  $II_1$  can be seen as an imperfect procedure.

$II_2$  is an instance of constitutive reasoning. The idea of the library regulations is that applying them defines which (legal) consequences hold in a particular situation. There is no outside criterion, other than the library regulations themselves, that allows us to evaluate the legal implications as far as the library is concerned. Hence, the reasoning process can be seen as a pure procedure.

The difference between epistemical and constitutive reasoning has implications for what principles do or do not hold in the reasoning process. Let us ask the question of whether some principle (like contraposition) holds in constitutive reasoning. The answer, of course, is that it depends on how the particular form of constitutive reasoning is defined. This definition needs not to be explicit. It may very well be that a certain type of informal reasoning has become common in a certain community, and that it is the researcher’s task to provide a formal model of this reasoning; this is essentially what happens in, for instance, AI & Law.

In epistemical reasoning, an outside criterion is available for determining whether the results are considered correct or not. The task of the reasoner is to perform its reasoning in such a way that the outcome approximates the objective criterion as closely as possible. In a certain sense, the presence of an objective criterion *forces* the reasoning process to become of a certain shape, in which certain properties (like contraposition) hold and other properties do not hold.

In constitutive reasoning, such an objective criterion is absent. For the community of reasoners, there is nothing that forces their reasoning process to become of a certain shape. In essence, the reasoners rely only on their own opinions and intuitions regarding what such a reasoning process should look like and which properties it should adhere to. Wason’s card experiment, however, makes clear that a large group of people has

difficulties with the principle of contraposition; it should therefore not come as a surprise that, when no outside constraint or criterion is present that *forces* its validity, the type of unreflective reasoning that a group of people comes up with does not necessarily sanction contraposition.

## References

- [1] P. Baroni, M. Giacomin, and G. Guida. Scc-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence*, 168(1-2):165–210, 2005.
- [2] Pietro Baroni and Massimiliano Giacomin. Comparing argumentation semantics with respect to skepticism. In *Proc. ECSQARU 2007*, pages 210–221, 2007.
- [3] Pietro Baroni and Massimiliano Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.
- [4] T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
- [5] Ph. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128 (1-2):203–235, 2001.
- [6] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.
- [7] M.W.A. Caminada. *For the sake of the Argument. Explorations into argument-based reasoning*. Doctoral dissertation Free University Amsterdam, 2004.
- [8] M.W.A. Caminada. Semi-stable semantics. In P.E. Dunne and T.J.M. Bench-Capon, editors, *Computational Models of Argument; Proceedings of COMMA 2006*, pages 121–130. IOS Press, 2006.
- [9] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, 77:321–357, 1995.
- [10] M.L. Ginsberg. Ai and nonmonotonic reasoning. In D. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, pages 1–33. Clarendon Press, Oxford, 1994.
- [11] M. Goldszmidt, P. Morris, and J. Pearl. A maximum entropy approach to nonmonotonic reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:220–232, 1993.
- [12] J. C. Hage. *Reasoning with rules; an essay on legal reasoning and its underlying logic*. Kluwer Academic Publishers, Dordrecht, 1997.
- [13] J.C. Hage. A theory of legal reasoning and a logic to match. *Artificial Intelligence and Law*, 4:199–273, 1996.
- [14] A.R. Lodder. *Dialaw, On Legal Justification and Dialog Games*. PhD thesis, University of Maastricht, 1998.
- [15] Sanjay Modgil. An abstract theory of argumentation that accommodates defeasible reasoning about preferences. In *Proc. ECSQARU 2007*, pages 648–659, 2007.
- [16] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.
- [17] J. Pearl. Epsilon-semantics. In S.C. Shapiro, editor, *Encyclopedia of Artificial Intelligence*, pages 468–475. Wiley, 1992.
- [18] H. Prakken. *Logical Tools for Modelling Legal Argument. A Study of Defeasible Argumentation in Law*. Law and Philosophy Library. Kluwer Academic Publishers, Dordrecht/Boston/London, 1997.
- [19] H. Prakken and G. Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artificial Intelligence and Law*, pages 331–368, 1996.
- [20] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7:25–75, 1997.
- [21] J. Rawls. *A Theory of Justice*. Oxford University Press, Oxford, 2000. revised edition.
- [22] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.
- [23] J.R. Searle. *Speech Acts, An Essay in the Philosophy of Language*. Cambridge University Press, 1969.
- [24] J.R. Searle. A taxonomy of illocutionary acts. In *Expression and Meaning*, pages 1–29. Cambridge University Press, 1979.
- [25] J.R. Searle. *Intentionality, An Essay in the Philosophy of Mind*. Cambridge University Press, 1983.

# Political Engagement Through Tools for Argumentation

Dan CARTWRIGHT<sup>a,1</sup> and Katie ATKINSON<sup>a</sup>

<sup>a</sup> *Department of Computer Science, University of Liverpool, UK.*

## **Abstract.**

In this paper we discuss the development of tools to support a system for e-democracy that is based upon and makes use of existing theories of argument representation and evaluation. The system is designed to gather public opinions on political issues from which conclusions can be drawn concerning how government policies are presented, justified and viewed by the users of the system. We describe how the original prototype has been augmented by the addition of well motivated tools to enable it to handle multiple debates and to provide analyses of the opinions submitted, from which it is possible to pinpoint specific grounds for disagreement on an issue. The tool set now supports both argumentation schemes and argumentation frameworks to provide representation and evaluation facilities. We contrast our system with existing fielded approaches designed to facilitate public consultation on political issues and show the particular benefits that our approach can bring in attempting to improve the quality of such engagement online.

**Keywords.** Tools for Supporting Argumentation, Reasoning about Action with Argument, Applications of Argumentation, e-Democracy, Argument Schemes.

## **1. Introduction**

The past few years have seen an increase in research to address some of the challenges posed by e-democracy. The reasons underpinning this drive are manifold. Firstly, there is the technological drive. With more members of the public having access to high speed, low cost internet connectivity, there is a desire to move traditional methods of government-to-public communication (and vice versa) online, to speed up and enhance such interactions. Subsequently, in a bid to mobilise the electorate in engagement with political issues, tools to encourage participation are desirable. Furthermore, the governing bodies can also exploit new technologies to support the provision, gathering and analysis of the public's contributions to political debate. With these aims in mind, we discuss a system for e-democracy that makes use of argumentation tools. The system, named Parmenides, was introduced in [2,3] and here we report on developments to the system that have substantially increased its functionality and analysis facilities, through the use of argumentation mechanisms. The tools developed are application driven, having arisen through the identification of issues presented by the domain that they have been built for, as opposed to being technology driven and thus searching for a suitable application.

---

<sup>1</sup>Correspondence to: Department of Computer Science, University of Liverpool, L69 3BX, UK. Tel.: +44 (0)151 795 4293; Fax: +44 (0)151 794 3715; E-mail: D.R.Cartwright@liverpool.ac.uk

The paper is structured as follows. In Section 2 we motivate the need for such a system by discussing particular existing approaches to e-democracy and their shortcomings. In Section 3 we briefly summarise our introductory work on the Parmenides system, then in Section 4 we describe the tools developed to meet the needs identified in the evaluation of the original prototype, plus the benefits that the tools bring. We finish in Section 5 with some concluding remarks and ideas for future developments.

## 2. Current Trends in e-Democracy

E-democracy and the encouraging of public participation in democratic debate is currently seen as an important obligation for governments, both national and local. The range of Internet-based tools used to support e-democracy vary in their purpose and implementation. One particular noteworthy example is the e-consultation systems described in [5], which are used to support and encourage young people in Scotland to participate in democratic decision making. Other examples are the numerous tools that are based on the use of web-based discussion boards, e.g. as discussed in [6]. As noted in [5], although such discussion boards can indeed encourage participation and debate, they generally provide no structure to the information gathered, so opportunity for analysis of the opinions is limited<sup>2</sup>. Generally in such systems there is often a problematic trade-off between the quality of contribution that users can make and the usability of the systems.

More recently e-petitions have become a popular mechanism. One such example is a site for filing petitions to government which has been running in the UK since November 2006 at <http://petitions.pm.gov.uk/>. This site enables users to create, view and sign petitions. The motivation behind the use of such petitions on this site is stated as making “it easy to collect signatures, and it also makes it easier for us to respond directly using email”. Whilst these petitions may facilitate signature collection and subsequent response, and be simple to use, the quality of engagement is questionable due to the numerous problems suffered by this method of communication. Firstly, e-petitions as used here are simply, as the name suggests, electronic versions of paper petitions. Whilst making petitions electronic may increase their visibility by exploiting current favoured methods of communication, they still suffer from the same shortcomings as paper versions. The most significant of these is conflation of a number of issues into one stock statement. By means of clarification, consider the following e-petition, taken from the aforementioned website, which proposes to “repeal the Hunting Act 2004”.

Petitioners know that The Hunting Act 2004: has done nothing for animal welfare; threatens livelihoods in the longer term; ignores the findings of Lord Burn's Enquiry; gives succour to animal rights extremists; is based on political expedience following the Prime Minister's unconsidered response on the television programme Question Time in 1999; is framed to persecute a large minority who support a traditional activity; does not command popular support in the country except amongst the uninformed and mal-advised.

By the deadline for the closure of the petition (November 2007) it had attracted 43,867 signatures. Once such a petition is closed it is then passed on to the relevant of-

<sup>2</sup>We note that tools for argument visualisation exist, e.g. Argunet (<http://www.argunet.org/>): our focus here is not primarily on argument visualisation, but argument gathering, representation and evaluation.

ficials or government department for them to provide a response. The website states that “Every person who signs such a petition will receive an email detailing the government’s response to the issues raised”. However, we do not believe that such a stock response can appropriately address each signatory’s individual concerns with the issue. This is precisely because the statement of the issue covers numerous different points and motives for disagreement, whilst those signing the petition will have more particular concerns. What is desirable is that the government response is not itself a stock reply, but customised to individual concerns. If we consider the example petition given above, we can see that the requested repeal is grounded upon numerous different elements:

- disputed facts, e.g. (i) that the act ignores the findings of The Burns Enquiry (which, prior to the act, investigated the impact of fox hunting and the consequences of a ban); and (ii) that public support for the act is low;
- the bad consequences that have followed from implementation of the act, e.g. (i) that there is an absence of improvement in animal welfare; (ii) that the act supports the activities of animal rights extremists; and (iii) that the act poses a long term threat against livelihoods;
- the misaligned purposes that the act promotes, e.g. (i) the unjustified persecution of those who support hunting with dogs; and (ii) the political gain of the Prime Minister following the introduction of the act.

So, in signing the above e-petition it can only be assumed that the signatory agrees wholeheartedly with the objections raised in the statement. This makes it easy to oversimplify the issues addressed in the petition. It is more likely that individuals support repeal of the act, but for differing reasons. For example, a user may agree that the act does not improve animal welfare and gives succour to animal rights extremists, but may disagree that it threatens livelihoods. Thus, signing such a petition is an “all-or-nothing” statement with no room for discriminating between (or even acknowledging) the different reasons as to why people may wish the act to be repealed. Furthermore, it may be that the petition does not cover all of the objections that can be made against the act and there are no means by which individuals can add any other objections they may have.

The above issues in turn have consequences for how an analysis of the petition is conducted and how the results are responded to by the government. After a petition closes the response is analysed quantitatively in terms of the number of signatures it attracted. Information is available from the e-petitions website as to the ranking of petitions, in terms of their relative popularity. Therefore, analysts could see which issues appear to be of most importance to members of the public who engage with the system. A response to the petition is then drafted which attempts to clarify the government’s position on the matter and respond to the criticisms made in the petition. However, since, as noted above, there is no means by which to discriminate between the particular reasons presented as to why the petition was endorsed, the stock response is not likely to adequately address each individual’s particular concerns. Any answer, therefore, can only be “one size fits all” and so fail to respond to the petitioners as individuals. Furthermore, it may be that the response given does not focus on the most contentious part of the issue, due to the amalgamation of the individual arguments. We thus believe it would be beneficial to recognise the different perspectives that can be taken on such issues where personal interests and aspirations play a role, a point recognised in philosophical works such as [7], and as used in approaches to argumentation, as discussed in [1].

It follows from the issues identified above that such e-petitions do not provide a fine grained breakdown of the justification of the arguments presented, and thus any responses sent to users cannot accommodate the individual perspectives internal to the arguments. In Section 4 we show how recent development of Parmenides increasingly attempts to address these points, but first we provide a brief description of the original system.

### 3. Overview of Parmenides

As originally described in [3], Parmenides is an online discussion forum that is based upon a specific underlying model of argument. It is intended as a forum by which the government is able to present policy proposals to the public so users can submit their opinions on the justification presented for the particular policy. The justification for action is structured in such a way as to exploit a specific representation of persuasive argument based on the use of argument schemes and critical questions, following Walton [8]. Argument schemes represent stereotypical patterns of reasoning whereby the scheme contains presumptive premises that favour a conclusion. The presumptions can then be tested by posing the appropriate critical questions associated with the scheme. In order for the presumption to stand, satisfactory answers must be given to any such questions that are posed in the given situation. The argument scheme used in the Parmenides system is one to represent persuasive argument in practical reasoning. This scheme has previously been described in [1], and it is an extension to Walton's *sufficient condition scheme for practical reasoning* [8]. The extended scheme, called AS1, is intended to differentiate several distinct notions conflated in Walton's 'goal'. AS1 is given below:

AS1 In the current circumstances R, we should perform action A, which will result in new circumstances S, which will realise goal G, which will promote some value V.

Instantiations of AS1 provide *prima facie* justifications of proposals for action. By making Walton's goal more articulated, AS1 identifies further critical questions that can be posed to challenge the presumptions in instantiations of AS1, making sixteen in total, as against the five of [8]. Each critical question can be seen as an attack on the argument it is posed against and examples of such critical questions are: "Are the circumstances as described?", "Does the goal promote the value?", "Are there alternative actions that need to be considered?". The full list of critical questions can be found in [1].

Given this argument scheme and critical questions, debates can then take place between dialogue participants whereby one party attempts to justify a particular action, and another party attempts to present persuasive reasons as to why elements of the justification may not hold or could be improved. It is this structure for debate that forms the underlying model of the Parmenides system, whereby a justification upholding the action proposed for the particular debate is presented to users of the system in the form of argument scheme AS1. Users are then led in a structured fashion through a series of web pages that pose the appropriate critical questions to determine which parts of the justification the users agree or disagree with. Users are not aware (and have no need to be aware) of the underlying structure for argument representation but it is, nevertheless, imposed on the information they submit. This enables the collection of information which is structured in a clear and unambiguous fashion from a system which does not require users to gain specialist knowledge before being able to use it.

In [2] it was suggested that Parmenides could be integrated with other methods of argument representation and evaluation, in particular Value-based Argumentation Frameworks [4]. Since all opinions submitted to Parmenides are written to a back-end database, the arguments can be organised into an Argumentation Framework to evaluate which elements of the justification have the most persuasive force. Tools to support this extension have been implemented, along with numerous other substantial features to expand and enhance the functionality of the system. We describe these tools in the next section.

#### 4. Additional Tools to Support Parmenides

In this section we provide details of a number of implemented tools to support Parmenides. These tools can be broadly categorised as: 1) Tools to allow the system to collect opinions on different topics of debate; 2) Tools for the analysis of data collected from opinions submitted through the website; 3) Tools for demographic profiling of users.

The original implementation of Parmenides [3] was based on the 2003 Iraq War debate. The tools have been implemented to facilitate modelling of other debates. One particular example is based on fox hunting, as described in the e-petition format earlier in this paper, and it poses the question “Should the fox hunting ban be repealed?”. For comparison purposes we will use this debate as a running example throughout the rest of the paper. The debate appears on the Parmenides system at:

<http://cgi.csc.liv.ac.uk/~parmenides/foxhunting/>

The initial statement instantiating AS1 for this debate is presented to the user as follows:

**In the current situation:** The ban affects the livelihoods of those who make a living from hunting, Less humane methods of controlling fox population have been introduced, The ban prejudices those who enjoy hunting with dogs, The ban ignores the findings of a government enquiry, The ban gives succour to animal rights extremists.

**Our goals are:** Create more jobs in the countryside, Reduce the need for less humane methods of fox control, Remove the prejudice against people who enjoy fox hunting, Take heed of the government enquiry, Withdraw support for animal rights extremists.

**This will achieve:** Creating more jobs promotes prosperity, Reducing the need for inhumane methods of fox control promotes animal welfare, Removing the prejudice against those who enjoy fox hunting promotes equality, Taking heed of the government enquiry promotes consistency, Withdrawing support for animal rights extremists promotes tolerance.

If we consider this debate as presented in the e-petition discussed earlier, we can see that Parmenides is easily able to represent the arguments as put forward there<sup>3</sup>. As per its implementation with the Iraq War debate, the system first asks the user whether he agrees or disagrees with the initial position. Those who disagree with it are then presented with a series of the appropriate critical questions, tailored to this specific debate, in order to uncover the specific element of the justification that they disagrees with. For example,

---

<sup>3</sup>Within our representation of this debate we have excluded the argument given in the e-petition that is based upon the Prime Minister’s appearance on the television programme Question Time, since this is a specific point with no clarification of the underlying details given. It would, of course, be possible to include this argument in our representation, and any other such ones excluded, if it were so desired.

the following question (instantiating CQ3) is posed to users to confirm their agreement with the achievement of goals by the action, as given in the initial position:

**Do you believe that repealing the ban would achieve the following?:**

Create more jobs in the countryside.

Reduce the need for less humane methods of fox control.

Remove the prejudice against people who enjoy fox hunting.

Take heed of a government enquiry.

Withdraw support for animal rights extremists.

After users have submitted their critique of the initial position, they are given the opportunity to construct their own position by choosing the elements of the position, from drop-down menus, that best reflect their opinion. We acknowledge that in restricting the users' choices to options given in drop down menus we constrain their freedom to express their opinions fully. However, such a trade-off, whilst not entirely desirable, is necessary if we are to capture overlapping opinions on an issue and automate their collection and analysis. Furthermore, allowing for input of entirely free text responses would increase both the risk of abuse of the system and the administration work involved in managing and responding to data collection of such opinions, including identifying and collating semantically equivalent but syntactically different responses. In an attempt to afford some element of increased expressivity to users, we do provide limited facilities to allow them to enter free text, with a view to drawing attention to elements of the debate that they believe have been excluded. Such elements could be considered for inclusion by the system's administrator, were a large number of similar omissions to be uncovered.

#### *4.1. Creating a New Debate*

The debate described above is just one example of how Parmenides is implemented to model a different topic of debate to that given in the original version of the system. To make it simple to add new debates we have implemented the 'Debate Creator', which enables administrators of the system to create new debates for presentation on the forum. The application allows administrators to input the parameters of a debate and it outputs the associated PHP webpages, plus a database source file. The Debate Creator requires little technical knowledge on the part of the administrators: they do not need to have knowledge of website and database design, nor specify the page ordering and layout necessary for the system to operate correctly. They are only required to understand the different elements that constitute the argument scheme used.

To create a new debate using this tool, the administrator must enter details of both the content of the debate, i.e. all the elements of the initial position and the drop-down menu options available for providing an alternative position, and details of the supporting technology, i.e. details of the SQL database host to which the data will be written. The data entered is used to create PHP webpage files, SQL database files, and data files necessary for analysis of the debate, without the need for any coding on the part of the administrator. This ensures that the debate remains internally consistent, requiring each data item to be entered only once (and the data is then propagated appropriately), and the format of the debate remains consistent with other debates on the forum. To aid usability, the Debate Creator provides support to ensure that all details of new debates are entered in the correct format. This consists of small help buttons next to each input box, which the user can click on to get more information about, and examples of, the input required.

## 4.2. Analysis Facilities

In order to analyse the opinion data submitted by users of the Parmenides website, a Java-based application has been implemented that analyses the arguments through the use of Argumentation Frameworks (AFs). The application consists of two analysis tools: the ‘Critique statistics analysis tool’ and the ‘Alternative position analysis tool’. Both tools retrieve user submitted opinions from the database and analyse them using Argumentation Frameworks to enable administrators to view the conclusions that can be drawn from the analysis. We discuss each tool in turn.

The ‘Critique statistics analysis tool’ analyses the individual critiques that users have given of the initial position of the debate and computes a set of statistics that reflect the analysis. The arguments are automatically translated into an AF graph representation that is displayed and annotated with the relevant statistics, allowing the administrator to easily see which element of the initial position users agree or disagree with most. Figure 1 shows an example of the tool being used to analyse the results of the fox hunting debate.

Within the argumentation frameworks displayed, the initial position is broken down into a number of sub-arguments, one for each of the social values promoted in the initial statement. For example, in the fox hunting debate the initial position presented actually comprises five separate arguments<sup>4</sup>, consisting of the relevant statements supporting each of the five social values promoted by the initial position. This can be seen in Figure 1 where the individual arguments are presented in tabular format along the top of the screen. In the centre of the framework is a large node containing the sub-argument for the currently selected social value, which in this case is ‘Prosperity’. The five arguments comprising the initial position are as follows:

- The ban affects the livelihoods of those who make a living from hunting. Repealing the ban will create more jobs in the countryside. Creating more jobs promotes Prosperity. Prosperity is a value worth promoting.
- Less humane methods of controlling fox population have been introduced. Repealing the ban will reduce the need for less humane methods of fox control. Reducing inhumane methods of fox control promotes animal welfare. Animal welfare is a value worth promoting.
- The ban prejudices those who enjoy hunting with dogs. Repealing the ban will remove the prejudice against people who enjoy hunting. Removing the prejudice against those who enjoy hunting promotes equality. Equality is a value worth promoting.
- The ban ignores the findings of a government enquiry. Repealing the ban will take heed of a government enquiry. Taking heed of a government enquiry promotes consistency. Consistency is a value worth promoting.
- The ban gives succour to animal rights extremists. Repealing the ban will withdraw support for animal rights extremists. Withdrawing support for animal rights extremists promotes tolerance. Tolerance is a value worth promoting.

In the analysis, each of these sub-arguments is displayed as a separate AF. The sub-arguments are broken down further into the individual elements (circumstances, goals,

---

<sup>4</sup>In using this particular debate we intend to show that our representation can capture the arguments used in the corresponding e-petition: we make no claim about the comprehensive coverage of all arguments related to the debate and acknowledge that there may be other numerous relevant aspects not included here.

values and purpose) that constitute the sub-argument, and each element is then assigned to a node in the AF. Nodes are also assigned to the ‘counter-statement’ of each element, the counter-statement effectively being one that is the opposite of the individual element in question. For example, consider the bottom right hand branch of the AF in Figure 1. Here, the statement for the particular element under scrutiny, the goal element, is “Repealing the ban will create more jobs in the countryside”. The counter-statement is simply its opposite: “Repealing the ban will not create more jobs in the countryside”. Through the critical questioning users are asked to say whether they agree or disagree with each positive statement, hence the need for the AF to show the opposing arguments. Each node is labelled with the number of users that agree with the representative statements.

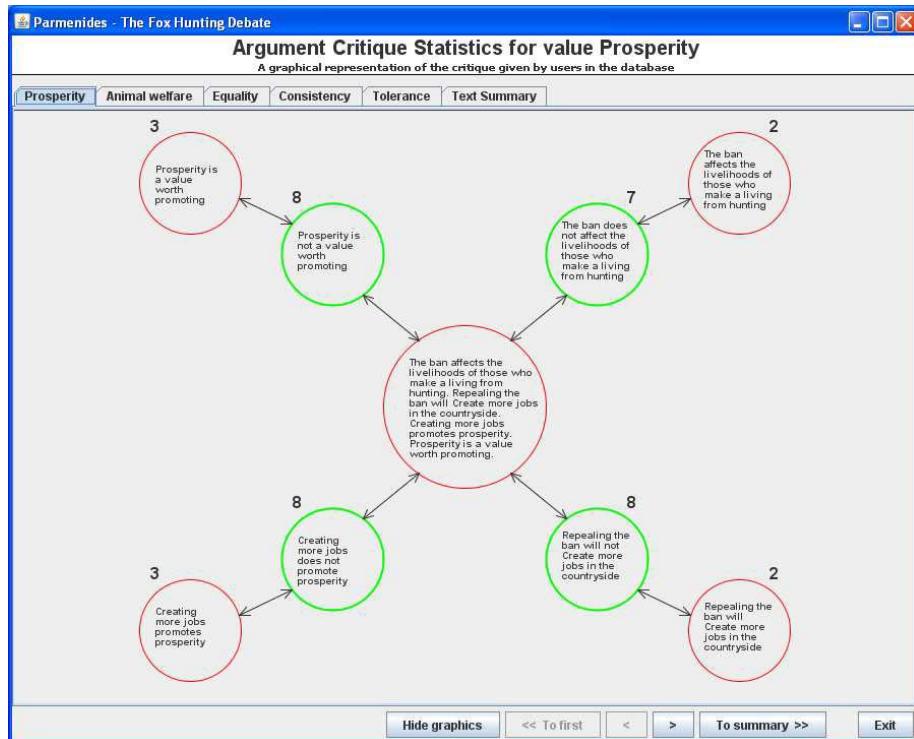


Figure 1: Critique statistics analysis framework for the fox hunting debate.

The critique statistics represented in the AF can be evaluated to determine the level of support for the various elements of the initial position. In the AF, for each sub-argument we define attacks between each element statement and its counter-statement. Defeat is determined by considering the statistics associated with each statement and its counter-statement. If more users have expressed their agreement with the counter-statement for a particular element, then the node representing the positive statement for the element is said to be defeated.

The attack relations are present not only between the individual elements and their counter-statements, but also between the counter-statements and the full sub-argument (the one represented in the central node of the AF). Therefore whenever a counter-statement has more support than its corresponding positive statement, the attack of the

counter-statement on the central node succeeds and the full sub-argument is deemed to be unjustified. This can be seen in the screenshot in Figure 1.

As described above, each AF has one branch for each element of the sub-argument. Consider again the branch in the bottom right hand corner of Figure 1, which relates to the ‘Consequence’ element. In this small-scale example, eight users agree with the counter-statement whereas only two users agree with the statement supporting this value. Therefore, the attack of the counter-statement on both the positive statement for the element and the sub-argument itself succeed. This is indicated with a green outline being given to the node representing the ‘winning’ counter-statement and a red outline to the nodes representing the statement and the sub-argument, which are both defeated.

The tool also provides a textual summary of the statistics, allowing the user to obtain an overview of support for various elements of the initial position. The textual summary may be a preferable form of analysis when the initial position of a debate promotes a large number of social values that make it difficult to visualise the numerous associated AF graphs. The textual summary can be used to easily determine which particular element of the argument is most strongly disagreed with. Consider the example presented in Figure 2, showing the statistics for our example debate. From these statistics we can easily determine that the social value with least overall support is ‘equality’ with an average of only 5% agreement with the statements supporting the value. Towards the bottom of the textual summary, the overall agreement with circumstances, goals, purposes and values is also displayed. In this case, circumstances and goals have least support amongst the users. The administrator would thus be able to pass on the conclusions to the relevant body who may consider making clear the evidence given for the current circumstances being as stated, or reviewing the relevance of the particular circumstances that are presented in the original position.

The advantage of this analysis of opinions over the e-petition representation described earlier in the paper is obvious: we can now see exactly which *particular* part of the debate is disagreed with by the majority of users. This can be either in the form of agreement with overall social values that are promoted by the position and their supporting statements, or in the form of aggregated statistics for each element of the position (circumstances, values, goals and purpose). In the case of the government e-petitions, once a petition is closed, an email is sent to all signatories to attempt to explain why the government policies critiqued are in place and operate as they do. However, as noted in Section 2, it is undoubtedly very difficult for the government to adequately address each person’s concerns since they are not aware of users’ specific reasons for disagreeing. Parmenides, in contrast, would allow the administrator to see which parts of the justification are most strongly disputed and hence enable any such correspondence to be more targeted so that it addresses the appropriate disputed part(s) of the government’s position. We could now modularise responses into paragraphs to be included or omitted, according to the concerns expressed by particular addressees.

The second analysis tool is the ‘Alternative position analysis tool’. This tool analyses the positions submitted by users as an alternative to the initial position. The positions are represented as Value-based Argumentation Frameworks (VAFs) [4] to represent and evaluate positions that promote different values. The Java program automatically constructs the VAF by assigning a node in the framework to each unique social value specified in the alternative positions constructed by users. Also assigned to each node is a list of actions that are specified in alternative positions that promote the value represented

by the node. One VAF is constructed for each of the social values promoted by the initial position of the debate, and within each framework a node is assigned to this social value. All nodes representing social values promoted by alternative positions attack the value promoted by the initial position, within each framework.

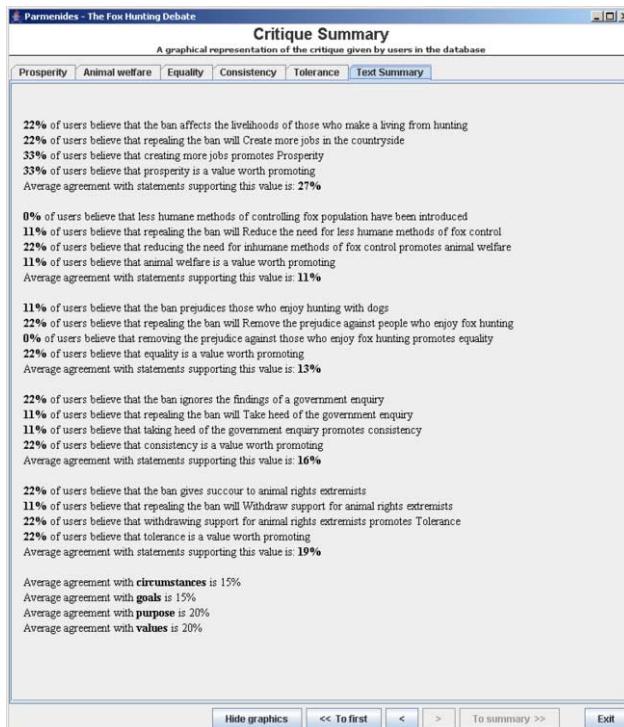


Figure 2. Critique statistics summary.

The Alternative position analysis tool can be used to obtain a subset of actions, from those submitted within positions that are alternative to the initial one, which can be considered ‘justifiable’ actions to carry out. We obtain the set of justifiable actions by applying a ranking over the values that appear in each VAF. For example, consider the screenshot in Figure 3. This shows a VAF based on the social value ‘Prosperity’ from the initial position of our example debate, with the central node representing the value. Surrounding and attacking this node are nodes representing social values promoted by alternative positions, as subscribed to by users. At this point we show the arguments without evaluating their relative status, thus we do not know which attack(s) succeed.

In order to determine whether or not an attack succeeds, following the definition of VAFs in [4], a ranking can be applied over the values in order to determine precedence. To obtain the ranking, the administrator is presented with an interface which allows him to input an ordering on all the values included in the initial position (the value ranking could alternatively be implemented as a “vote” on users endorsing the values). Once the ranking has been given, the arguments are evaluated as follows: if an argument attacks another whose value has a lesser ranking, the attack succeeds; if an argument attacks another whose value has a higher ranking, the attack fails; if an argument attacks another whose value is the same as that of the attacker, the attack succeeds.

Once the value ranking has been applied, the VAF is updated to show the status of the arguments according to the given ranking. Those arguments that are defeated have their associated nodes outlined in red and those outlined in green are not defeated. The actions which promote the values represented by the green nodes can then be considered ‘justifiable’ actions to carry out, since they withstood the critiques applied given the value ranking, and any one may be justifiably chosen to execute. This set of actions is output on screen, concluding presentation of the analysis data.

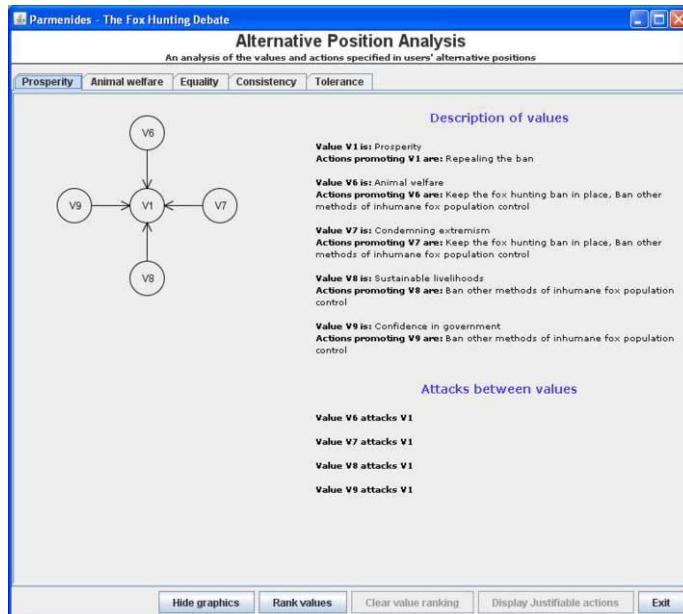


Figure 3. VAF showing competing alternative arguments.

#### 4.3. Profiler

So far we have described the Parmenides system, its use in an e-government context, and the tools implemented to analyse the data submitted. We now briefly describe an additional tool that allows for demographic profiling of the system’s users. The Parmenides Profiler allows users to create an account, which is then associated with every debate on which they submit an opinion, as well as a demographic profile which the user may optionally complete with their personal details, such as education, marital status, lifestyle, etc. The interface provides a list of live debates currently on the forum, which can be altered using the Parmenides Profiler Admin Portal, a PHP webpage for Parmenides administrators. The profiler system interface can be viewed at: <http://cgi.csc.liv.ac.uk/~parmenides/Profiler/>

Although there is currently no functionality to analyse the data collected by the Profiler, there is plenty of scope for developing such tools. For example, tools could be created to see if certain responses to particular arguments within a debate are popular with a certain demographic of people. It would also be possible to analyse all opinions of each individual user to determine whether they always believe that the same values are worth promoting, for example. Alternatively the government, or other body interested in

the debate, may wish to filter the opinions of certain demographics. For example, they may wish to see the difference in opinion between males and females or different age groups. The same could be done for policies that affect certain sections of the population, e.g. road policies, where it may be useful to analyse the difference in opinion submitted by those who hold driving licences to those who do not. We hope to incorporate such demographic profiling facilities into future work on the Parmenides system.

## 5. Concluding Remarks

In this paper we have described the development of a prototype system and suite of tools to support e-democracy, which make use of specific methods for representing and reasoning with arguments. These tools were motivated by support for the following features: facilities to enable multiple debates on different topics to be presented within a common format; a tool to enable administrators to create their own debates, on any topic, from which an associated website and database are dynamically created to present the debate; analysis facilities that allow the arguments submitted to be represented as Argumentation Frameworks, from which statistical information concerning a breakdown of support for the arguments can be gathered, and value based arguments assessed; and, a profiler system to enable demographic profiling of users from their responses to multiple debates.

Future work will be to extend the profiler tool to include analysis facilities, and investigate methods to increase and better handle free text input of users' own opinions. We also intend to investigate how we could make use of other argumentation schemes to expand and enhance the range of arguments and types of reasoning that are used in the system. Finally, and most importantly, we intend to conduct large scale field tests to validate the effectiveness of the system, investigations for which are currently underway.

To summarise, we believe that the current developments of the Parmenides system that we have described here begin to address some of the shortcomings of other systems that do not allow for a fine-grained analysis of the arguments involved in a debate, and this is strengthened by the use of existing methods of argument representation and evaluation that are themselves hidden from the users, ensuring the system is easy to use.

## References

- [1] K. Atkinson. *What Should We Do?: Computational Representation of Persuasive Argument in Practical Reasoning*. PhD thesis, Department of Computer Science, Liverpool University, 2005.
- [2] K. Atkinson. Value-based argumentation for democratic decision support. In P. E. Dunne and T. Bench-Capon, editors, *Computational Models of Argument, Proceedings of COMMA 2006*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 47–58. IOS Press, 2006.
- [3] K. Atkinson, T Bench-Capon, and P. McBurney. PARMENIDES: Facilitating deliberation in democracies. *Artificial Intelligence and Law*, 14(4):261–275, 2006.
- [4] T. Bench-Capon. Persuasion in practical argument using value based argumentation frameworks. *Journal of Logic and Computation*, 13 3:429–48, 2003.
- [5] A. Macintosh, E. Robson, E. Smith, and A. Whyte. Electronic democracy and young people. *Social Science Computer Review*, 21(1):43–54, 2003.
- [6] Ø. Sæbø and H. Nilsen. The support for different democracy models by the use of a web-based discussion board. In R. Traunmüller, editor, *Electronic Government*, LNCS 3183, pages 23–26. Springer, 2004.
- [7] J. R. Searle. *Rationality in Action*. MIT Press, Cambridge, MA, USA, 2001.
- [8] D. N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.

# A Computational Model of Argumentation in Everyday Conversation: A Problem- Centred Approach

Jean-Louis DESSALLES

TELECOM ParisTech

46 rue Barrault - 75013 Paris - France

*dessalles@enst.fr*

**Abstract.** Human beings share a common competence for generating relevant arguments. We therefore hypothesize the existence of a cognitive procedure that enables them to determine the content of their arguments. The originality of the present approach is to analyse spontaneous argument generation as a process in which arguments either signal problems or aim at solving previously acknowledged problems.

**Keywords.** Argumentation, conversation, abduction, cognition.

## 1. Modelling Spontaneous Argumentation

Human beings are all expert in argument generation. Modelling natural argumentation, as it occurs in spontaneous conversation, is important for two reasons at least. First, it is a source of inspiration for argumentation system design, which can choose to imitate it to some extent. Second, as automatically generated arguments are intended to human hearers or readers, it is important to understand the way the latter generate relevant utterances for each other. This paper offers a new model of spontaneous argument generation.

Argumentation represents the major part of spontaneous speech. Our own measures (Table 1) show a distribution of conversational genres, with figures assessed through a sampling method. The corpus we used is composed of 17 hours of family conversation. Conversation proper, which excludes utilitarian (more or less ritualized) speech, occupies more than 70% of the time, and argumentation amounts to 74% of conversation time.

There are strong constraints on every argumentative move in daily conversation, as ill-formed arguments are promptly regarded as pathological and their author as socially inept [1]. We therefore hypothesize the existence of a Human Argumentative Procedure (HAP). This paper is aimed as an attempt to characterize central aspects of the HAP.

Most current approaches to argumentation involve computations over pre-existing ‘arguments’. The problem is then to find out the best argument among a set of more or less adequate moves. However, conversational topics are so varied that finding *one single* acceptable argument is often quite a difficult task. Finding a model for HAP consists in explaining how arguments are *computed*, not how they are merely selected

from memory. Human knowledge is potentially huge and hardly circumscribable. We must suppose that it is content-addressed only. This assumption contrasts with most models, in which memory can be randomly searched or scanned for available arguments, to check consistency or detect cycles.

Table 1: Distribution of conversational genres in a corpus of family conversations

Conversation	60 %	{	Argumentative discussions	74 %
Conversation (inaudible)	13 %		Narratives	19 %
Other (child screaming, songs)	5 %		Immediate events	7 %
Utilitarian (mainly negotiation about food offer)	11 %			
Empty	11 %			

Natural arguments are generated using only domain knowledge and common sense knowledge. This knowledge-to-argument (K2A) perspective excludes that arguments be manipulated as such or through relations like ‘attacks’ [2]. In a K2A approach, the very notion of argument emerges from the functioning of the argumentative procedure.

The HAP described here essentially consists in (1) detecting some local incompatibility in the participants’ current beliefs and desires, and (2) attempting to resolve this incompatibility. Several authors have noticed that reasoning and argumentation are governed by conflicting beliefs and desires [3] and that argumentation aims at lowering the internal conflict [4]. Our approach differs mainly in the fact that all computations are supposed to be *local*. For instance, Pasquier *et al.* [4] carry out global operations like summing over all constraining relations. We consider such operations to be unrealistic as human knowledge is content-addressed and cannot be scanned.

Our enterprise is to define a minimal argumentative procedure to *generate* (and not merely select) arguments using a K2A approach. The difficulty is to reconstruct real conversations using only domain knowledge. This problem still constitutes a challenge for A.I. and cognitive modelling.

We first introduce a few basic concepts underlying the model: cognitive conflict, strength, abduction. Then we outline our model of HAP, and illustrate how it works on examples. We conclude by mentioning how the model has been implemented.

## 2. Cognitive Conflicts and Abduction

The argument generation procedure starts with the detection of a cognitive conflict, and stops when this conflict, or the subsequent ones that may come out during the resolution process, have disappeared, or when no solution can be found. Resolution relies on the central mechanism of abduction.

A *cognitive conflict* is detected whenever a given proposition is assigned two opposite *strengths*. We call *strength* of a state of affairs the intensity with which this state of affairs is believed or wished by participants. Strengths are negative in case of

disbelief or avoidance. At each step  $t$  of the planning procedure, a function  $v_t(T)$  is supposed to provide the strength of any proposition  $T$  on demand. When the strength of  $T$  is neither given nor inferred,  $v_t(T) = 0$ . A cognitive conflict is a situation in which a same proposition  $T$  is successively given two opposite strengths:  $v_t(T) \cdot v_{t+1}(T) < 0$ . The conflict, noted  $(T, n_1) \uparrow (T, n_2)$ , has an intensity, given by the product  $I = -n_1 n_2$ . Note that cognitive conflicts are internal to the agents; they are not supposed to be objective. More important, *cognitive conflicts do not oppose persons*, but representations.

*Abduction* is central to problem-solving [5], to diagnosis reasoning [6] and more generally to human intelligence [7]. It is also essential to the argumentative procedure proposed here. For the sake of simplicity, causal links are supposed to be explicitly given to the model, and abduction is performed by using causal links backwards. Abduction from  $E$  using the causal clause  $(C_1 \& C_2 \& \dots \& C_n) \rightarrow E$  returns the weakest cause in the clause, *i.e.*  $\text{Argmin } v_i(C_i)$ . This is, of course, a gross simplification. Further developments of the model could involve procedures to perform Bayesian abduction or sub-symbolic simulations to make the abduction part more plausible. We distinguish *diagnostic abduction* from *creative abduction*. The former returns only actual (*i.e.* observed) causes, whereas the latter may return any cause from the chosen clause.

### 3. Resolving Cognitive Conflicts

The argumentative procedure is inherently problem-based: It is launched as soon as the current proposition  $T$  creates a conflict (we may consider  $T$  as the last event observed or the last input in working memory).

- (a) Conflict: Consider the conflict  $(T, -n_1) \uparrow (T, n_2)$ , with  $n_1 > 0$  and  $n_2 > 0$ . There may be a certain threshold  $I_0$ , depending on the context, below which the conflict is ignored. If  $I = n_1 n_2 > I_0$ , the resolving procedure goes as follows.
- (b) Propagation: Perform diagnostic abduction from  $T$  ( $T$  is不相信 or unwanted with strength  $n_1$ ). If successful, it returns an actual cause  $C_i$  of  $T$ . If  $0 < v_i(C_i) < n_1$ , the cognitive conflicts propagates to its cause: Make  $v_{i+1}(C_i) = -n_1$ , and go through step (b) anew with the cognitive conflict  $(C_i, -n_1) \uparrow (C_i, v_i(C_i))$ . However, if  $v_i(C_i) \leq 0$ , the conflict is solved through negation by suggesting  $\neg C_i$ .

In the following conversation, adapted from [8], we see how cognitive conflict propagation leads participants to produce arguments.

*C- How did you get – I mean how did you find that side of it, because...*

*A- Marvellous*

*C- You know some people say that... that driving a car across a ferry is the devil of a job [. . .] well I'll tell you the sort of thing I've heard, I mean every summer, you see stories of tremendous queues at the...*

*D- But they're people who haven't booked*

The initial cognitive conflict is about driving a car across the Channel, which is presented as ‘marvellous’ by A and D and ‘the devil of a job’ by C. At some point, C propagates the conflict onto its actual cause: the mention of ‘tremendous queues’. D did not have to wait in these queues, so he propagates the new conflict onto an actual cause for being in such queues: not having booked, which happens to have a negative

strength in A and D's case. The conflict thus vanishes, as the strength inherited from 'marvellous' is negative too. We can see how the content of the three arguments ('driving a car across the ferry is the devil of a job', 'you see stories of tremendous queues', 'but they're people who haven't booked') results from conflict detection and propagation.

- (c) Reparation: If propagation fails, negate  $T$  to form the *counterfactual*  $\neg T$  ( $\neg T$  is believed or wanted with strength  $n_1$ ) and perform creative abduction. If successful, it returns a (possible) cause  $C_i$  of  $\neg T$ . If  $-n_1 < \nu_t(C_i) < 0$ , make  $\nu_{t+1}(C_i) = n_1$  and go to step (b) with the cognitive conflict  $(\neg C_i, -n_1) \uparrow (\neg C_i, -\nu_t(C_i))$ . If  $\nu_t(C_i) \geq 0$ , suggest  $C_i$ ; if  $C_i$  is an action and is feasible, simulate its execution by making its consequences actual and reset  $\nu_{t+1}(C_i)$  to 0; then observe the resulting situation and restart the whole procedure.

Consider the following conversation (original in French). R, S and their friends want to project slides on a white door, as they have no screen.

[The projector would be ideally placed on the shelves, but it is unstable]

R- *Can't you put the projector there [on the desk]?*

S- [...] *it will project on the handle. That will be nice!*

R- *Put books underneath. But can't you tilt it?*

S- *It will distort the image*

R initial suggestion (put the projector on the desk) is motivated by the instability of the projector, which creates a cognitive conflict. The conflict propagates to its cause. Then reparation occurs: the problematic term (projector on shelves) is negated, and an action is found that realises this counterfactual: remove the projector from the shelves. The procedure goes on, with an alternation of conflict detection, propagation and reparation (Table 2).

**Table 2.** Covert and overt argumentative moves

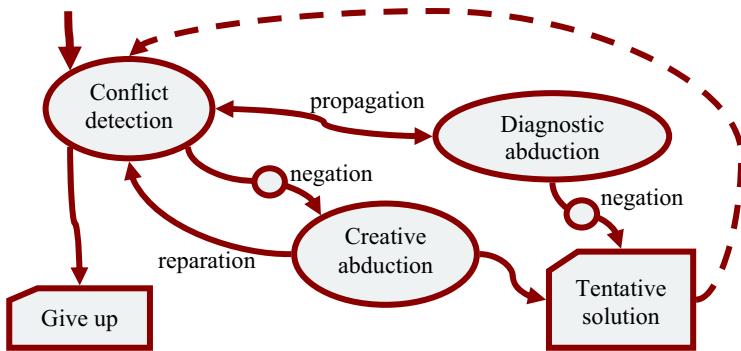
Argumentative move	Procedure phase
projector unstable	<i>Conflict detection</i>
projector on the shelves	<i>Propagation</i>
remove the projector from the shelves	<i>Repair</i>
image no longer on the door	<i>Conflict detection</i>
"Can't you put the projector there?" [on the desk]	<i>Repair</i>
"I'll project on the handle."	<i>Conflict detection</i>
the projector is horizontal	<i>Propagation</i>
"Put books underneath."	<i>Repair</i>
"But can't you tilt it?"	<i>Repair</i>
"It will distort the image."	<i>Conflict detection</i>

- (d) Failure: When reparation fails, make  $\nu_{t+1}(T) = n_1$  ( $T$  is thus marked as resisting resolution with strength  $n_1$ ) and redo the whole procedure.

At the end of the preceding dialogue, the strength  $n_1$  of the image distortion ( $D$ ) is inherited from the strength of tilting the projector (through (c)), which is itself inherited from the strength of not having the image projected on the handle. When  $D$  is observed, it conflicts with the desire  $n_2$  of having a non-distorted image. The conflict reads:  $(D, n_1) \uparrow (D, -n_2)$ . If  $n_1 > n_2$  and there is no actual cause weaker than  $n_2$  for  $D$ , propagation fails. If there is no way to produce  $\neg D$ , reparation fails as well. One proceeds through step (d) and  $D$  is stored with the new strength  $n_1$ . This leaves the situation with the unresolved conflict  $(D, -n_1) \uparrow (D, n_1)$ .

- (e) Giving up: The system exits the resolution procedure when it is blocked and the strength value hierarchy does not change.

Figure 1 summarizes the whole argumentation generation procedure.



**Figure 1.** The argument generation procedure

#### 4. Conclusion

We implemented the model in a Prolog programme. For such an implementation to remain plausible, the size of the programme must be kept minimal. The above procedure is realized with less than 130 Prolog goals (15 clauses), excluding display, domain knowledge and state transitions when an action is performed. This amount is five times less than previous attempts [9]. The programme is able to produce the same arguments as those observed in a variety of real dialogues, using a small number of steps. This performance should not be underestimated. Usually, planning procedures consider many useless possibilities, and unlike humans, base their choice on multiple evaluations. The challenge of the approach is not only to produce the correct argumentation, but also to produce it in a reasonable number of steps and with a minimally complex procedure.

The current model of HAP may still be improved. For instance, ‘negation’ and ‘abduction’ are still called twice in the procedure. We may think of an even simpler version of the procedure, but it is still to be discovered.

One important result suggested by the model is that human argumentation can be achieved without central embedding. Though argumentative dialogues most often end up as balanced trees of arguments, the procedure that generates them is only right-

recursive. The reason is that the procedure constructs trees of arguments by exploring the web of constraints in a non-deterministic and often redundant way.

The model is conceived to offer a tentative plausible image of cognitive processes underlying argument generation. It does not aim at technical efficiency. If used to process specific task-driven dialogues, it could prove as inefficient as would be a novice in comparison with an expert. However, the model may prove technically helpful when limited knowledge is available to utter arguments that will nevertheless appear relevant. It may be also useful to understand the relevance of users' arguments.

The pursued objective is scientific rather than technical. We consider our approach as a promising step toward better understanding of human spontaneous argumentation. The current limitations of the model are due to the extreme simplification of the knowledge made available to the system, which consists of explicit causal rules. The good side of it is that the argument generation procedure is general, simple and systematic, and offers a plausible, though still partial, image of the human spontaneous argumentative ability.

## References

- [1] Polanyi, L. (1979). So What's the point?. *Semiotica*, 25 (3), 207-241.
- [2] 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, 321-357.
- [3] Bratman, M. E. (1990). What is intention? In P. R. Cohen, J. L. Morgan & M. E. Pollack (Eds.), *Intentions in communication*, 15-32. Cambridge, MA: MIT press.
- [4] Pasquier, P., Rahwan, I., Dignum, F. P. M. & Sonenberg, L. (2006). Argumentation and persuasion in the cognitive coherence theory. In P. Dunne & T. Bench-Capon (Eds.), *First international conference on Computational Models of Argumentation (COMMA)*, 223-234. IOS Press.
- [5] Magnani, L. (2001). Abduction, reason and science - Processes of discovery and explanation. New York: Kluwer Academic.
- [6] Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial Intelligence*, 32, 57-95.
- [7] Hobbs, J. R., Stickel, M. & Martin, P. (1993). Interpretation as abduction. *Artificial Intelligence*, 63 (1-2), 69-142.
- [8] Crystall, D. & Davy, D. (1975). *Advanced Conversational English*. Londres: Longman.
- [9] Dessalles, J-L. (1990). The simulation of conversations. In T. Kohonen & F. Fogelman-Soulie (Eds.), *Proceedings of the Cognitiva-90 Symposium*, 483-492. Amsterdam: North Holland.  
[http://www.enst.fr/~jld/papiers/pap.conv/Dessalles\\_90072003.html](http://www.enst.fr/~jld/papiers/pap.conv/Dessalles_90072003.html)

# Towards argumentation-based contract negotiation

Phan Minh DUNG<sup>a</sup>, Phan Minh THANG<sup>a</sup>, Francesca TONI<sup>b,1</sup>

<sup>a</sup> Asian Institute of Technology, Bangkok, Thailand

<sup>b</sup> Department of Computing, Imperial College London, UK

**Abstract.** We present an argumentation-based approach to contract negotiation amongst agents. Contracts are simply viewed as abstract transactions of items between a buyer agent and a seller agent, characterised by a number of features. Agents are equipped with beliefs, goals, and preferences. Goals are classified as either *structural* or *contractual*. In order to agree on a contract, agents engage in a two-phase negotiation process: in the first phase, the buyer agent decides on (a selection of) items fulfilling its structural goals and preferences; in the second phase, the buyer agent decides on a subset of the items identified in the first phase fulfilling its contractual goals and preferences. The first phase is supported by argumentation-based decision making taking preferences into account.

**Keywords.** Negotiation, Decision-making, Contracts

## Introduction

We present an argumentation-based approach to contract negotiation amongst agents. Contracts are simply viewed as abstract transactions of items between a buyer agent and a seller agent, characterised by a number of features. Agents are equipped with beliefs, goals, and preferences. Beliefs are represented as an assumption-based argumentation framework. Goals are literals classified as either *structural* or *contractual*, depending on whether they are about structural, static properties of the item the agents aim at agreeing on to form the contract, or whether they are about features subject to negotiation leading to the agreement of a contract. Preferences are given by numerical rankings on goals.

In order to agree on a contract, agents engage in a two-phase negotiation process: in the first phase, the buyer agent decides on (a selection of) items fulfilling its structural goals and preferences; in the second phase, the buyer agent decides on a subset of the items identified in the first phase fulfilling its contractual goals and preferences. The outcome of the second phase is a set of possible contracts between the buyer and the seller. The first phase is supported by argumentation-based decision making with preferences.

We ground our proposed framework upon a concrete “home-buying” scenario, whereby the buyer agent is looking for a property to buy, and the seller has a number of properties to sell. In this scenario, the structural goals are features of a property such as its location, the number of rooms, etc, and the contractual goals are the price of the

---

<sup>1</sup>Corresponding Author: F. Toni, Imperial College London, UK ; E-mail: ft@doc.ic.ac.uk.

property, the completion date for the sale, etc. A contract is simply the agreement on a concrete property and on a number of features fulfilling all “preferred” goals (according to the given preferences).

The paper is structured as follows. In section 1 we give background on assumption-based argumentation, the form of argumentation we adopt to support the agents’ decision-making during negotiation. In section 2 we define the form of contracts we use. In section 3 we define the internal structure and format of the agents in our framerwork, based upon assumption-based argumentation and preferences on goals. In section 4 we outline a two-phase negotiation process used by the agents to agree on a contract. In section 5 we discuss relationship to related work and conclude.

## 1. Background

This section provides the basic background on assumption-based argumentation (ABA), see [3,5,6,4] for details.

An ABA framework is a tuple  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\neg} \rangle$  where

- $(\mathcal{L}, \mathcal{R})$  is a *deductive system*, consisting of a language  $\mathcal{L}$  and a set  $\mathcal{R}$  of inference rules,
- $\mathcal{A} \subseteq \mathcal{L}$ , referred to as the set of *assumptions*,
- $\overline{\neg}$  is a (total) mapping from  $\mathcal{A}$  into  $\mathcal{L}$ , where  $\overline{x}$  is referred to as the *contrary* of  $x$ .

We will assume that the inference rules in  $\mathcal{R}$  have the syntax  $l_0 \leftarrow l_1, \dots, l_n$  (for  $n \geq 0$ ) where  $l_i \in \mathcal{L}$ . We will refer to  $l_0$  and  $l_1, \dots, l_n$  as the *head* and the *body* of the rule, respectively. We will represent  $l \leftarrow$  simply as  $l$ . As in [5], we will restrict attention to *flat* ABA frameworks, such that if  $l \in \mathcal{A}$ , then there exists no inference rule of the form  $l \leftarrow l_1, \dots, l_n \in \mathcal{R}$ , for any  $n \geq 0$ .

An *argument* in favour of a sentence  $x$  in  $\mathcal{L}$  supported by a set of assumptions  $X$  is a (backward) deduction from  $x$  to  $X$ , obtained by applying backwards the rules in  $\mathcal{R}$ .

In order to determine whether a conclusion (set of sentences) should be drawn, a set of assumptions needs to be identified providing an “acceptable” support for the conclusion. Various notions of “acceptable” support can be formalised, using a notion of “attack” amongst sets of assumptions whereby  $X$  attacks  $Y$  iff there is an argument in favour of some  $\overline{x}$  supported by (a subset of)  $X$  where  $x$  is in  $Y$ . Then, a set of assumptions is deemed

- *admissible*, iff it does not attack itself and it counter-attacks every set of assumptions attacking it;
- *preferred*, iff it is maximally admissible.

We will refer to a preferred set of assumptions as a *preferred extension* of the given ABA framework. We will use the following terminology:

- a preferred extension of  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\neg} \rangle \cup \{a\}$ , for some  $a \in \mathcal{A}$ , is a preferred extension of  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\neg} \rangle$  containing  $a$ ;
- given a preferred extension  $E$  and some  $l \in \mathcal{L}$ ,  $E \models l$  stands for “there exists a backward deduction for  $l$  from some  $E' \subseteq E$ ”;
- given a preferred extension  $E$  and some  $L \subseteq \mathcal{L}$ ,  $E \models L$  stands for  $E \models l$  for all  $l \in L$ .

## 2. Contracts

We will assume a set of (at least two) agents *Agents*, a set of *Items*, and a set *Attributes* of attributes for the elements of *Items*. Each attribute is associated with a domain of possible values: for each  $a \in Attributes$ , this domain is indicated as  $D(a)$ .

A *contract* is defined simply as a transaction between (some of the) agents, playing different roles. This transaction is characterised by an item and a number of features, possibly including technical aspects and cost of the item. Concretely, we will assume that these features are assignments of values to attributes, for the given item. For simplicity, in this paper we will restrict attention to contracts between two agents playing the role of buyer and seller. Formally, a contract is a tuple  $\langle Buyer, Seller, Item, Features \rangle$  where

- $Buyer, Seller \in Agents$ ,  $Buyer \neq Seller$ , representing the buyer and seller in the contract;
- $Item \in Items$ ;
- $Features$  is a set of assignments of values to (some of the) attributes:  $Features = \bigcup_{a \in X} \{a = v_a\}$  for some  $X \subseteq Attributes$  and, for each  $a \in X$ , some  $v_a \in D(a)$  representing the value of attribute  $a$  for *Item*.

Given a contract with *Features* for attributes in some  $X$ , we will consider attributes not in  $X$  as irrelevant, in the sense that their values could be any in their domain without altering the worth of the contract.

Attributes may take any number of values. For Boolean attributes, with domain  $\{\text{true}, \text{false}\}$ , we will represent assignments of attributes to *true* simply by means of the attributes, and assignments of attributes to *false* simply by means of the negation of the attributes. So, for example,  $\{a_1 = \text{true}, a_2 = \text{false}, a_3 = \text{true}\}$  will be represented as  $\{a_1, \neg a_2, a_3\}$ .

In the remainder of the paper, for simplicity, we will assume that  $Agents = \{\beta, \sigma\}$ , with  $\beta$  the buyer and  $\sigma$  the seller in every contract.

As an illustrative example, throughout the paper we will use a “home-buying” scenario whereby two agents, a home buyer and an estate agent, engage in negotiations for the purchase of a property. In this scenario, the given set of agents is  $\{h\_buyer, e\_agent\}$ , representing a home-buyer and an estate agents respectively ( $h\_buyer$  is a concrete instance of  $\beta$  and  $e\_agent$  is a concrete instance of  $\sigma$ ). Also, in this scenario, items are properties for sale, and the attributes include *exchange\_date* (representing the date when a non-refundable deposit is paid by the buyer to the seller and contracts are exchanged), *completion\_date* (representing the date when the final payment is made and the property changes hands) and *price*. An example contract is

$\langle h\_buyer, e\_agent, house_1, \{completion\_date = 10/03/08, price = \$300K\} \rangle$  indicating that  $h\_buyer$  is committed to purchasing  $house_1$  at the price of  $\$300K$  and with agreed completion date for the deal on 10 March 2008.

## 3. Agents’ internals

An agent is characterised by its own goals, preferences over goals, and beliefs. Beliefs and goals are expressed in a given logical language  $\mathcal{L}$  consisting of literals (we do not impose any restriction on  $\mathcal{L}$ , for example it may not include negation and, if it does, it

may not be closed under negation). This language is shared amongst agents. The literals in  $\mathcal{L}$  representing possibly desirable properties for agents are called *goal literals*<sup>2</sup>. For example, a goal literal may be that of having a comfortable house (*comfortable\_house*), or a house with at least 3 rooms (*number\_of\_rooms*  $\geq 3$ ), or a house costing less than \$ 450K (*price*  $\leq \$450K$ ). Goal literals may be of two kinds: those concerning the attributes of the items to be bought (for example location, number of rooms, foundations, building permits etc) and goals concerning the contractual features of such items (for example price, time of completion, deposit etc). Beliefs may be about: the items to be traded, norms and conventions governing the agents' behaviour, and issues agents are uncertain about.

Formally, any agent  $\alpha$  is defined as a tuple  $\langle G_\alpha, B_\alpha, P_\alpha \rangle$  consisting of

- a *goal-base*  $G_\alpha \subseteq \mathcal{L}$  describing the agent's own goals, and consisting of two disjoint subsets:  $G_\alpha = G_\alpha^{\text{struct}} \cup G_\alpha^{\text{contr}}$ , where  $G_\alpha^{\text{struct}}$  are the *structural goals* and  $G_\alpha^{\text{contr}}$  are the *contractual goals*
- a *belief-base*  $B_\alpha$  describing the agent's beliefs
- a *preference-base*  $P_\alpha$  describing the agent's preferences over its goals

For simplicity, in this paper we will assume that the seller agent  $\sigma$  only has contractual goals (namely  $G_\sigma^{\text{struct}} = \{\}$ ).

We will omit the subscript  $\alpha$  from the bases when clear from the context.

The goal-base describes important features of the item the buyer is looking for. The preference-base allows to rank different items according to preferences on their features. The belief-base of both buyer and seller needs to include information about concrete items that can become part of contracts (for example that a given property has 5 rooms), relevant to contractual goals, about norms and conventions used by the agents during the negotiation of the contracts (for example that a seller moving overseas is going to be in a rush to sell, or that a recently-built property with council approval is likely to be safe), and about uncertainties the agents may have during this process (for example that the asking price for a property is too high). Syntactically:

- preferences over goals are expressed by assigning positive integers to them, namely the preference-base is a mapping from the goal-base to the set of natural numbers, ranking the goals so that the higher the number assigned to a goal, the more important the goal
- the belief-base is an ABA framework  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \neg \rangle$  where
  - \*  $\mathcal{R} = \mathcal{R}_i \cup \mathcal{R}_n \cup \mathcal{R}_c$ , where
    - \*  $\mathcal{R}_i$  represents information about concrete items to be traded
    - \*  $\mathcal{R}_n$  represents (defeasible) norms
    - \*  $\mathcal{R}_c$  represents information related to contractual goals
  - \*  $\mathcal{A} = \mathcal{A}_d \cup \mathcal{A}_c \cup \mathcal{A}_u$  where
    - \*  $\mathcal{A}_d$  consists of assumptions representing (decisions about) items for transactions, for example *house<sub>1</sub>*, *house<sub>2</sub>*
    - \*  $\mathcal{A}_c$  represents "control" assumptions related to defeasible norms (see below)

---

<sup>2</sup>We abuse notation here in the sense that something may be desirable for one agent but not for others.

- \*  $\mathcal{A}_u$  contains assumptions representing the uncertainties about attributes of items to be traded, e.g. whether a given property has a completion certificate

Note that  $\mathcal{R}_i$ ,  $\mathcal{R}_c$ , and  $\mathcal{A}_d$  can be obtained directly from information about items to be traded. The rest of the belief base of an agent corresponds to item-independent beliefs held by the agent, and need to be “programmed” into the agent.

Deciding whether or not to start a negotiation about an item in  $\mathcal{A}_d$  depends on how this item is evaluated according to the assumptions in  $\mathcal{A}_u$ . For example, the lack of a completion certificate is an indication that the house may not be safe. We intend here that the will to (dis)confirm assumptions in  $\mathcal{A}_u$  will drive information-seeking steps in the contract negotiation process.

As a simple example of buyer in our running scenario,<sup>3</sup>  $h\_buyer$  may consist of

- $G_{h\_buyer}^{struct} = \{own\_garden, number\_of\_rooms \geq 3, safe\}$   
 $G_{h\_buyer}^{contr} = \{completion\_date < 31/05/08, price < \$450K\}$
- $P_{h\_buyer}(own\_garden) = 2, P_{h\_buyer}(number\_of\_rooms \geq 3) = 2$   
 $P_{h\_buyer}(completion\_date < 31/05/08) = 3$   
 $P_{h\_buyer}(price < \$450K) = 4$   
 $P_{h\_buyer}(safe) = 5$
- $B_{h\_buyer}$  is  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \neg \rangle$  where

\*  $\mathcal{R} = \mathcal{R}_i \cup \mathcal{R}_n \cup \mathcal{R}_c$  and

$$\mathcal{R}_i = \{ number\_of\_rooms = 5 \leftarrow house_1,$$

$$price = \$400K \leftarrow house_2,$$

$$seller\_in\_chain \leftarrow house_2 \}$$

$$\mathcal{R}_n = \{ safe \leftarrow council\_approval, asm_1,$$

$$\neg safe \leftarrow weak\_foundations, asm_2,$$

$$council\_approval \leftarrow completion\_certificate, asm_3,$$

$$long\_time\_not\_sold \leftarrow price\_too\_high, asm_4,$$

$$seller\_in\_rush \leftarrow seller\_moves\_overseas, asm_5 \}$$

$$\mathcal{R}_c = \{ long\_time\_not\_sold \leftarrow house_1,$$

$$seller\_moves\_overseas \leftarrow house_2,$$

$$quick\_completion \leftarrow seller\_moves\_overseas,$$

$$completion\_date < now + 60days \leftarrow quick\_completion \}$$

\*  $\mathcal{A} = \mathcal{A}_d \cup \mathcal{A}_u \cup \mathcal{A}_c$  and

$$\mathcal{A}_d = \{house_1, house_2\}$$

$$\mathcal{A}_c = \{asm_1, asm_2, asm_3, asm_4, asm_5\}$$

$$\mathcal{A}_u = \{\neg price\_too\_high, price\_too\_high, \neg seller\_in\_rush, \neg council\_approval\}$$

---

<sup>3</sup>We omit to exemplify the seller for lack of space.

- \*  $\overline{house_1} = house_2, \overline{house_2} = house_1,$   
 $\overline{asm_1} = \neg safe, \overline{asm_2} = safe, \overline{asm_3} = \neg council\_approval,$   
 $\overline{asm_4} = short\_time\_not\_sold, \overline{asm_5} = \neg seller\_in\_rush,$   
 $\overline{price\_too\_high} = \neg price\_too\_high,$   
 $\neg \overline{price\_too\_high} = price\_too\_high,$   
 $\neg \overline{seller\_in\_rush} = seller\_in\_rush,$   
 $\neg \overline{council\_approval} = council\_approval.$

Note that there are different kinds of uncertainties. Some are directly related to the structural properties of items to be traded, like the lack of a council approval. Others are related either to the contractual properties of the items, like *price\_too\_high*, or to the behaviors of the other agent, like *seller\_in\_rush*. Note also that assumptions in  $\mathcal{A}_u$  are not ordinary assumptions, in that they would not be normally assumed by the agent unless explicit information is obtained. For example, the agent would not ordinarily assume that the seller is in a rush: it will want to check this. Indeed, assumptions in  $\mathcal{A}_u$  are meant to be used to start information-seeking dialogues in the negotiation process. This kind of dialogues will be ignored in this paper.

The assumptions in  $\mathcal{A}_c$  are used to reflect the defeasible nature of the corresponding rules and their potential to give rise to an inconsistency. These control assumptions can also be useful to resolve conflicts between conflicting information, e.g. originating from different sources of information. For example, the first rule above could be replaced by

$$\begin{aligned} number\_of\_rooms = 5 &\leftarrow house_1, asm_{11} \\ number\_of\_rooms = 4 &\leftarrow house_1, asm_{12} \end{aligned}$$

with  $asm_{11}, asm_{12}$  additional assumptions and  $\overline{asm_{11}} = number\_of\_rooms = 4$  and  $\overline{asm_{12}} = number\_of\_rooms = 5$ . This would reflect that the agent has been exposed to two conflicting pieces of information (possibly from two different sources of information), that *house<sub>1</sub>* has both 5 and 4 rooms, and would need to decide (by selecting one of  $asm_{11}$  or  $asm_{12}$ , or neither) which of the two it will make use of.

#### 4. Negotiation process

The decision making of a buyer can be structured into two phases. In the first phase the agent evaluates the items that are available, according to their attributes, to determine whether and how they satisfy its needs. In our running example, the agent would have to decide which houses satisfy its goals about location, safety etc. In the second phase, a negotiation will be conducted with the seller for items (e.g. houses) that have passed the first phase. These phases would benefit from information-seeking, but we ignore this here for simplicity. We focus instead on the decision-making mechanisms needed to support the two phases.

The principle for decision making for both phases is that higher-ranked goals should be pursued at the expense of lower-ranked goals, and thus choices enforcing higher-ranked goals should be preferred to those enforcing lower-ranked goals.

Choices in the first phase are items (e.g. houses) that are available for transactions. Choices in the second phases are possible deals that the buyer and seller could struck (e.g. prices, deposits etc).

Choices are compared based on how they satisfy the goals. A *goal state* (or simply a *state* for short) may be seen as a set of goal literals. Abstractly, this set is intended to be the set of all goals satisfied by a choice. We will see some concrete realisations of this abstract notion in sections 4.2 and 4.3 below. But first, in section 4.1, we will provide a generic way of comparing goal states by taking into account preferences amongst goal literals in them.

#### 4.1. Comparing Choices

As an example of the decision making principle given earlier, consider goals  $g_0, g_1, g_2$  in the goal-base  $G$  such that  $P(g_0) = P(g_1) = 2$  and  $P(g_2) = 1$ , where  $P$  is the agent's preference-base. Consider states  $s_1 = \{g_0, g_1\}$ ,  $s_2 = \{g_0, g_2\}$ ,  $s_3 = \{g_1, g_2\}$ . Then,  $s_1$  is preferred to both  $s_2, s_3$  whereas  $s_2, s_3$  are incomparable and thus equally preferred. Formally:

**Definition 4.1** Let  $s, s'$  be states. We say that  $s$  is *preferred to*  $s'$ , denoted by  $s \sqsupseteq s'$ , iff

1. there exists a goal  $g$  that is satisfied in  $s$  but not in  $s'$ , and
2. for each goal  $g'$ , if  $P(g') \geq P(g)$  and  $g'$  is satisfied in  $s'$  than  $g'$  is also satisfied in  $s$ .

It follows:

**Proposition 4.1** The preference relation  $\sqsupseteq$  is a partial order.

Often decisions need to be made even though the satisfaction of some goals is undetermined. For example, our buyer may want to buy a home that is situated in a quiet neighbourhood. But one of the properties on offer is located in an area where a project to build a new airport is under consideration by the government, and there are strong arguments for and against the airport. Some other property also under the buyer's consideration does not have council approval, and so may be unsafe. Deciding which of these two properties to buy amounts to comparing the preference over two sets of states. For example, in the case of the first property, a state with the airport being built and a state without the airport being built. This uncertainty is represented by allowing two assumptions *airport* and  $\neg$ *airport* in  $\mathcal{A}_u$ . Due to the presence of uncertainties, comparing items means comparing sets of states. A possible notion for this comparison is the following:

**Definition 4.2** Let  $S$  be a nonempty set of states. The *min-state* of  $S$ , denoted by  $\text{min}(S)$ , is a state such that for each goal  $g \in G$ ,  $g$  is satisfied in  $\text{min}(S)$  iff  $g$  is satisfied in each state in  $S$ . Let  $S, S'$  be sets of goal states.  $S$  is said to be *minmax-preferred* to  $S'$  if  $\text{min}(S)$  is preferred to  $\text{min}(S')$ .

#### 4.2. First phase: Decision making for the buyer

The items to be chosen (e.g. houses) are represented by assumptions in  $\mathcal{A}_d$ . In this setting, goal states are concretely defined as follows:

**Definition 4.3** A *structural (goal) state* is a maximal consistent <sup>4</sup> set of goal literals from  $G^{struct}$ .

Until the next section 4.3, we will refer to structural states simply as states.

Choices determine states as follows:

**Definition 4.4** Let  $s$  be a goal state,  $d \in \mathcal{A}_d$  and  $g \in G^{struct}$ . We say that

- $s$  is *credulously satisfied by*  $d$  if there is a preferred extension  $E$  of  $B \cup \{d\}$  such that  $E \models s$
- $g$  is *skeptically satisfied by*  $d$  if, for each preferred extension  $E$  of  $B \cup \{d\}$ ,  $E \models g$
- $s$  is *skeptically satisfied by*  $d$  if for each goal  $g \in G^{struct}$ ,  $g$  is skeptically satisfied by  $d$  iff  $g \in s$

It is not difficult to see that there is exactly one state that is skeptically satisfied by any given  $d$ .

For  $d \in \mathcal{A}_d$ , the *characteristic set of goal states* of  $d$ , denoted by  $CS_d$ , consists of all goal states  $s$  credulously satisfied by  $d$ .

**Definition 4.5** Given  $d_0, d_1 \in \mathcal{A}_d$ :

- $d_0$  is said to be *minmax preferred* to  $d_1$  if  $CS_{d_0}$  is minmax preferred to  $CS_{d_1}$
- $d_0$  is said to be *skeptically preferred* to  $d_1$  if the unique goal state that is skeptically satisfied by  $d_0$  is preferred (with respect to  $\sqsupseteq$ ) to the unique goal state that is skeptically satisfied by  $d_1$

The following result links our notion of (minmax) preference between states (the characteristic sets given by decisions) and our argumentation-based notion of (skeptical) preference between decisions:

**Proposition 4.2** Let  $d_0, d_1 \in \mathcal{A}_d$ .  $d_0$  is minmax preferred to  $d_1$  iff  $d_0$  is skeptically preferred to  $d_1$ .

Then, decision-making in the first-phase amounts to choosing any “most skeptically preferred” decision. We will refer to these decisions as the *most favored items*.

### 4.3. Second phase: How Should a Fair Negotiator Proceed ?

In this phase, buyer and seller make decisions by negotiation to agree on a contract. After the first phase, the buyer decides to start negotiation with the seller on (one of) the most favored items. Each agent (the buyer and the seller), ranks these items according to how they satisfy the contractual goals in  $G^{contr}$ . Note that the second phase is only concerned with the contractual goals - of both agents (the structural goals of the buyer have all been taken into account in the first phase, and we are assuming that the seller has no structural goals).

---

<sup>4</sup>A goal state is inconsistent if, for some atom  $g$ , it contains both  $g$  and its negation  $\neg g$ ; a goal state is consistent if it is not inconsistent. Note that we do not impose that  $\mathcal{L}$  is closed under negation, and in particular  $\mathcal{L}$  could be a set of atoms. In this special case, any set of goal atoms would be a goal state.

**Definition 4.6** A *contractual (goal) state* is a maximal consistent set of goal literals from  $G^{contr}$ .

In our home-buying example, a contractual state consists of a price, a deposit, time for completion and several add-ons items like washing-machines, curtains etc. We assume that the set of contractual states is finite and is known to both buyer and seller.

The preference of an agent  $\alpha$  (which may be the buyer  $\beta$  or the seller  $\sigma$ ) between contractual states can be represented as a total preorder  $\sqsupseteq_\alpha$ <sup>5</sup>, where, given contractual states  $t$  and  $t'$ ,  $t \sqsupseteq_\alpha t'$  states that  $t$  is preferred to  $t'$  (for  $\alpha$ ). As  $\sqsupseteq_\alpha$ , we can choose any pre-order consistent with the partial order  $\sqsupseteq$  obtained as in definition 4.1 for goals and preferences of  $\alpha$ .

For simplicity, we assume that both buyer and seller know each other's preferences between contractual states. We also assume that each agent  $\alpha$  possesses an evaluation function  $\lambda_\alpha$  that assigns to each structural state  $s$  a contractual state  $\lambda_\alpha(s)$  representing the "value" of  $s$ , such that if  $s$  is preferred to  $s'$  (as in definition 4.1) then  $\lambda_\alpha(s)$  is preferred to  $\lambda_\alpha(s')$ .

For the buyer agent  $\beta$ ,  $\lambda_\beta(s)$  represents the "reservation" value of  $s$ , i.e. the maximal offers the buyer could make (for the features affecting the contractual goals, that will determine the contract). For the seller agent  $\sigma$ ,  $\lambda_\sigma(s)$  represents the "reservation" value of  $s$ , i.e. the minimal offers the sellers could accept (for the features affecting the contractual goals, that will determine the contract).

From now on, we assume that the agents are negotiating about one of the most favored items characterized by structural state  $s$ . The possible deals (contracts) between the buyer and the seller are characterized respectively by the sets

- $PD_\beta = \{t \mid t \text{ is a contractual state and } t \sqsupseteq_\beta \lambda_\beta(s)\}$ ,
- $PD_\sigma = \{t \mid t \text{ is a contractual state and } t \sqsupseteq_\sigma \lambda_\sigma(s)\}$ .

If  $PD_\beta \cap PD_\sigma \neq \emptyset$  then a deal is possible. We define the *negotiation set* as  $NS = PD_\beta \cap PD_\sigma$ . We assume that the agents are rational in the sense that they would not accept a deal that is not Pareto-optimal, defined below:

**Definition 4.7** Let  $t, t'$  be contractual states. We say that:

- $t$  is strictly preferred to  $t'$  for agent  $\alpha$  if  $t \sqsupseteq_\alpha t'$  and  $t' \not\sqsupseteq_\alpha t$
- $t$  dominates  $t'$  if  $t$  is preferred to  $t'$  for both seller and buyer (i.e.  $t \sqsupseteq_\beta t'$  and  $t \sqsupseteq_\sigma t'$ ) and, for at least one of these agents,  $t$  is strictly preferred to  $t'$
- $t$  is Pareto-optimal if it is not dominated by any other contractual state

The agents bargain by successively putting forward offers. A negotiation is defined as a sequence of alternating offers and counter-offers from the negotiation set  $NS$  between the buyer and the seller. Offers and counter-offers are represented by contractual states. An agent could accept an offer or reject it and then make a counter-offer. We assume that our agents are honest and do not go back on their offers. This implies that when an agent makes a new offer, it should be at least as preferred to its opponent as the one it has made previously.

---

<sup>5</sup>A total preorder  $\sqsupseteq$  on a set  $T$  is a reflexive and transitive relation such that for each two elements  $t, t'$  from  $T$ , either  $t \sqsupseteq t'$  or  $t' \sqsupseteq t$ .

Reciprocity is a key principle in negotiation. There is no meaningful negotiation without reciprocity. An agent is said to adhere to the *principle of reciprocity* if, whenever the other agent has made a concession, it will reciprocate by conceding as well. We say that an agent *concedes* if its new offer is strictly preferred to its opponent than the one it made previously. Otherwise the agent is said to *stand still*. Agents do not have unlimited time for negotiation. Hence practical agents will terminate a negotiation when they see no prospect for a successful conclusion for it. This happens when both agents refuse to concede/reciprocate.

Offers and counter-offers may be seen as steps in a negotiation. A negotiation *terminates in failure* at step  $n$  if both agents stand still at steps  $n, n - 1, n - 2$ <sup>6</sup>. It is understood that a failure is worse than any agreement for both agents.

A negotiation *terminates successfully* when one of the agents accepts an offer. An agent  $\alpha$  accepts an offer from the other agent if it is preferred to  $\alpha$  to the one proposed by  $\alpha$  itself before. For example, consider the following negotiation between  $\beta$  and  $\sigma$ :

- Step 1:  $\beta$  puts forward an offer of 10.

Here, 10 can be seen as the price that  $\beta$  is prepared to pay for the item at stake, characterised by the contractual state  $s$ .

- Step 2:  $\sigma$  makes a counter-offer of 12.

Here, 12 can be seen as the price that  $\sigma$  is prepared to accept as a payment for the item.

- Step 3:  $\beta$  makes a further counter-offer of 11.

Namely,  $\beta$  increases the amount it is willing to pay for the item, in other words, it concedes.

- Step 4:  $\sigma$  makes a counter-offer of 11.

Namely,  $\sigma$  decreases the amount it is willing to accept for payment for the item, in other words, it concedes.

- Step 5:  $\beta$  accepts the offer.

Indeed, the offer by  $\sigma$  at step 4 is preferred to  $\beta$  (by being =) to the offer it made at step 3.

An agent  $\alpha$  is said to be *fair* if it adheres to the principle of reciprocity. Formally, this means that whenever  $\alpha$  has to move at step  $n$ , it will concede or accept if the number of concessions made by the other agent  $\bar{\alpha}$  up to step  $n - 1$  is more than the number of concessions made by  $\alpha$  up to step  $n - 1$ . Note that for ease of reference, we refer to the opponent of  $\alpha$  as  $\bar{\alpha}$ .

Due to the finiteness assumption of the set of contractual states, the negotiation set is also finite. Hence it is immediate that

**Theorem 4.1** Every negotiation terminates.

A strategy is defined as a mapping assigning to each history of negotiation an offer. We are now interested in strategies for fair agents that ensure an efficient and stable outcome in the sense of a Nash equilibrium.

---

<sup>6</sup>This means that when an agent stands still and in the next move its opponent also stands still then the first agent has to concede if it does not want to terminate the negotiation in failure.

**Definition 4.8** A contractual state  $t'$  is said to be a *minimal concession of agent  $\alpha$  wrt  $t$* , if  $t'$  is strictly preferred to  $t$  for  $\bar{\alpha}$  and for each contractual state  $r$ , if  $r$  is strictly preferred to  $t$  for  $\bar{\alpha}$  then  $r$  is preferred to  $t'$  for  $\bar{\alpha}$ .

An agent *concedes minimally* at step  $i$  if it offers at step  $i$  a contractual state  $t$  that is a minimal concession wrt the offer the agent made at step  $i - 2$ . The *minimal concession strategy* calls for agents

1. to start the bargain with their best state and
2. to concede minimally if the opponent has conceded in the previous step or it is making a move in the third step of the negotiation, and
3. to stand still if the opponent stands still in previous step.

Note that the third step in the negotiation has a special status, in that if no concession is made at that step the negotiation stops.

It is obvious that the minimal concession strategy adheres to the reciprocity principle. Hence the minimal concession strategy is permissible for fair agents.

It is not difficult to see

**Proposition 4.3** If both agents use the minimal concession strategy then they terminate successfully.

A strategy is said to be in *symmetric Nash equilibrium* if under the assumption that one agent uses this strategy the other agent can not do better by not using this strategy.

**Theorem 4.2** The minimal concession strategy is in symmetric Nash equilibrium.

**Proof Sketch** Let  $st$  be the minimal concession strategy and suppose that agent  $\alpha$  is using  $st$  and the other agent  $\bar{\alpha}$  is using  $st'$  that is different from  $st$ . If the negotiation terminates in failure then it is clear that the outcome is worse for  $\bar{\alpha}$  in comparison to the choice of using  $st$ . Suppose now that the negotiation terminates with an agreement  $t$ . Because  $\alpha$  uses the minimal concession strategy, if  $\bar{\alpha}$  stands still in one step, the negotiation will terminate in failure. Therefore we can conclude that there is no stand-still step according to  $st'$ . Let  $t_0$  be the agreement if both parties use the minimal concession strategy. We want to show that  $t_0 \sqsupseteq_{\bar{\alpha}} t$ . From the definition of the minimal concession strategy, it follows that no agent stands still in this negotiation. This implies that  $\bar{\alpha}$  in many steps makes a bigger concession than a minimal one. It follows then that  $t_0 \sqsupseteq_{\bar{\alpha}} t$ .

The Nash equilibrium of the minimal concession strategy means that when a fair agent is using the minimal strategy, the other agent is doing best by also using this strategy. In other word, the minimal concession strategy is an efficient and stable strategy for fair agents.

## 5. Conclusions

We have outlined a two-phase negotiation process whereby two agents, a buyer and a seller, aim at agreeing on an item fulfilling all “preferred” goals of the agents. These

goals are classified as structural and contractual. We have focused on covering the full negotiation life-cycle, from identifying items to be negotiated upon to conducting the actual negotiation for (contractual) features of these items. We have worked out how argumentation in general and assumption-based argumentation in particular can support the first phase. We have also proven several results on the outcome of the two-phase negotiation process, and defined a strategy for agents allowing them to achieve Nash equilibria.

We have made a number of simplifying assumptions. First, both agents are supposed to be honest and open. The seller agent is supposed to have no structural goals. We have ignored the need for information-seeking in both phases. In the future, we plan to extend this work by dropping these assumptions. We also plan to define a communication machinery to support our strategy and protocol.

We have illustrated our approach using a simple home-buying scenario. We believe that our approach could be fruitfully defined for other scenarios too, for example e-business scenarios like the ones studied in the ARGUGRID project<sup>7</sup>. We plan to study these other scenarios in the future.

The first phase is supported by a decision-making mechanism using argumentation and preferences. A number of such decision-making mechanisms exist, e.g. [8,10,9,2]. In this paper, we have provided an argument-based framework that can deal with decision making, uncertainties and negotiation but we have restricted ourselves only to a simple and ideal case where we assume that the agents are honest and open to each other.

The second phase could also be supported by argumentation. The use of argumentation here could be beneficial also to support resolution of disputes over contracts. We plan to explore this in the future.

Several works exist on argumentation-based negotiation [11]. For example, [12] propose a protocol and a communication language for dealing with refusals in negotiation. It would be useful to see how this protocol and communication language may be used to support the two-phase negotiation framework we have defined. Also, [1] presents an abstract negotiation framework whereby agents use abstract argumentation internally and with each other. Our framework instead is tailored to the specific case of contract negotiation and assumes a very concrete and structured underlying argumentation framework.

Our minimal concession strategy for fair agents is inspired by the monotonic concession protocol of [14], though it differs from it in significant ways. In our framework the agent moves alternatively where in [14] they move simultaneously. The condition for terminating the negotiation is also different. As a result, the minimal concession strategy is a symmetric Nash equilibrium in our framework while the corresponding strategy in [14] is not. Other work exists on deploying the minimal concession strategy within multi-agent systems, e.g. [7,13], looking at negotiation amongst multiple agents. In this paper we have considered just two agents, and focused instead on the full negotiation process, from the identification of issues to bargain about to the actual bargaining, thus linking argumentation-based decision making to the monotonic concession protocol.

---

<sup>7</sup>[www.argugrid.eu](http://www.argugrid.eu)

## Acknowledgements

This research was partially funded by the EU ARGUGRID project. We thank Kevin Ashley for some useful suggestions for future work.

## References

- [1] L. Amgoud, Y. Dimopoulos, and P. Moraitis. A unified and general framework for argumentation-based negotiation. In *Proc. AAMAS'2007*, 2007.
- [2] K. Atkinson and T. Bench-Capon. Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artificial Intelligence*, 171(10–15):855–874, 2007.
- [3] A. Bondarenko, P.M. Dung, R.A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1-2):63–101, 1997.
- [4] P.M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
- [5] P.M. Dung, R.A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170:114–159, 2006.
- [6] P.M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence, Special Issue on Argumentation in Artificial Intelligence*, 171(10–15):642–674, 2007.
- [7] U. Endriss. Monotonic concession protocols for multilateral negotiation. In P. Stone and G. Weiss, editors, *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, pages 392–399. ACM Press, May 2006.
- [8] A.C. Kakas and P. Moraitis. Argumentation based decision making for autonomous agents. In *Proc. AAMAS'03*, pages 883–890, 2003.
- [9] M. Morge and P. Mancarella. The hedgehog and the fox: An argumentation-based decision support system. In *Proc. ArgMAS*, 2007.
- [10] I. Rahwan and L. Amgoud. An argumentation-based approach for practical reasoning. In *Proc. AAMAS'06*, pages 347–354. ACM Press, 2006.
- [11] I. Rahwan, S. Ramchurn, N. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation-based negotiation. *The Knowledge Engineering Review*, 18(4):343 – 375, 2003.
- [12] J. van Veenen and H. Prakken. A protocol for arguing about rejections in negotiation. In *Proc. ArgMAS'06*, volume 4049 of *LNAI*, pages 138–153. Springer, 2006.
- [13] D. Zhang. Reasoning about bargaining situations. In *Procs AAAI-07*, pages 154–159, 2007.
- [14] G. Zlotkin and J. S. Rosenschein. Negotiation and task sharing among autonomous agents in cooperative domains. In *Proc. IJCAI*, pages 912–917, 1989.

# The Computational Complexity of Ideal Semantics I: Abstract Argumentation Frameworks

Paul E. DUNNE

*Department of Computer Science, The University of Liverpool, U.K.*

**Abstract.** We analyse the computational complexity of the recently proposed *ideal semantics* within abstract argumentation frameworks. It is shown that while typically less tractable than credulous admissibility semantics, the natural decision problems arising with this extension-based model can, perhaps surprisingly, be decided more efficiently than sceptical admissibility semantics. In particular the task of *finding* the unique maximal ideal extension is easier than that of *deciding* if a given argument is accepted under the sceptical semantics. We provide efficient algorithmic approaches for the class of *bipartite* argumentation frameworks. Finally we present a number of technical results which offer strong indications that typical problems in ideal argumentation are complete for the class  $P_{||}^{NP}$ : languages decidable by polynomial time algorithms allowed to make non-adaptive queries to an NP oracle.

**Keywords.** Computational properties of argumentation; Formalization of abstract argumentation; Computational complexity

## Introduction

The extension-based semantics defined by *ideal extensions* were introduced by Dung, Mancarella and Toni [11,12] as an alternative sceptical basis for defining collections of justified arguments in the frameworks promoted by Dung [10] and Bondarenko *et al.* [3].

Our principal concern in this article is in classifying the computational complexity of a number of natural problems related to ideal semantics in abstract argumentation frameworks, such problems including both *decision* questions and those related to the *construction* of the unique maximal ideal extension.<sup>1</sup> Thus,

- a. Given an argument  $x$  is it accepted under the ideal semantics?
- b. Given a set of arguments,  $S$ 
  - b1. Is  $S$  a subset of the maximal ideal extension?, i.e. without, necessarily, being an ideal extension itself.
  - b2. Is  $S$ , itself, an ideal extension?

---

<sup>1</sup>That every AF has a unique maximal ideal extension is shown by Dung *et al.* [12]

- b3. Is  $S$  the *maximal* ideal extension?
- c. Is the maximal ideal extension empty?
- d. Does the maximal ideal extension coincide with the set of all *sceptically accepted* arguments?
- e. Given an argumentation framework, construct its maximal ideal extension.

We obtain bounds for these problems ranging from NP, co-NP, and D<sup>P</sup>-hardness through to an exact P<sub>||</sub><sup>NP</sup>-completeness classification for the construction problem defined in (e). In the remainder of this paper, background definitions are given in Section 1 together with formal definitions of the problems introduced in (a)–(e) above. In Section 2, two technical lemmata are given which characterise properties of ideal extensions (Lemma 1) and of *arguments* belonging to the *maximal* ideal extension (Lemma 2). The complexity of decision questions is considered in Section 3 and in Section 4 an exact classification for the complexity of *finding* the maximal ideal extension is given. One consequence of this result is that (under the usual complexity-theoretic assumptions) *constructing* the maximal ideal extension of a given framework is, in general, easier than *deciding* if one of its arguments is sceptically accepted.

The results of Section 3 leave a gap between lower (hardness) bounds and upper bounds for a number of the decision questions. In Section 5 we present strong evidence that problems (a), (b1), (b3) and (c) are not contained in any complexity class strictly below P<sub>||</sub><sup>NP</sup>: specifically that all of these problems are P<sub>||</sub><sup>NP</sup>-hard via *randomized reductions* which are correct with probability approaching 1.

For reasons of limited space we will, generally, not present detailed proofs. Full proofs of all results may be found in Dunne [13].

## 1. Abstract Argumentation Frameworks

The following concepts were introduced in Dung [10].

**Definition 1** An argumentation framework (AF) is a pair  $\mathcal{H} = \langle \mathcal{X}, \mathcal{A} \rangle$ , in which  $\mathcal{X}$  is a finite set of arguments and  $\mathcal{A} \subset \mathcal{X} \times \mathcal{X}$  is the attack relationship for  $\mathcal{H}$ . A pair  $\langle x, y \rangle \in \mathcal{A}$  is referred to as ‘y is attacked by x’ or ‘x attacks y’. For  $R, S$  subsets of arguments in the AF  $\mathcal{H}(\mathcal{X}, \mathcal{A})$ , we say that  $s \in S$  is attacked by  $R$  – written  $\text{attacks}(R, s)$  – if there is some  $r \in R$  such that  $\langle r, s \rangle \in \mathcal{A}$ . For subsets  $R$  and  $S$  of  $\mathcal{X}$  we write  $\text{attacks}(R, S)$  if there is some  $s \in S$  for which  $\text{attacks}(R, s)$  holds;  $x \in \mathcal{X}$  is acceptable with respect to  $S$  if for every  $y \in \mathcal{X}$  that attacks  $x$  there is some  $z \in S$  that attacks  $y$ . A subset,  $S$ , is conflict-free if no argument in  $S$  is attacked by any other argument in  $S$ . A conflict-free set  $S$  is admissible if every  $y \in S$  is acceptable w.r.t  $S$  and  $S$  is a preferred extension if it is a maximal (with respect to  $\subseteq$ ) admissible set. A subset,  $S$ , is a stable extension if  $S$  is conflict free and every  $y \notin S$  is attacked by  $S$ . An AF,  $\mathcal{H}$  is coherent if every preferred extension in  $\mathcal{H}$  is also a stable extension.

A subset,  $S$ , is an ideal extension ([11, 12]) of  $\mathcal{H}$  if  $S$  is admissible and a subset of every preferred extension of  $\mathcal{H}$ . The AF,  $\mathcal{H}$  is cohesive if its maximal ideal extension coincides with the intersection of all preferred extensions of  $\mathcal{H}$ .

For  $S \subseteq \mathcal{X}$ ,

$$\begin{aligned} S^- &=_{\text{def}} \{ p : \exists q \in S \text{ such that } \langle p, q \rangle \in \mathcal{A} \} \\ S^+ &=_{\text{def}} \{ p : \exists q \in S \text{ such that } \langle q, p \rangle \in \mathcal{A} \} \end{aligned}$$

An argument  $x$  is credulously accepted if there is some preferred extension containing it;  $x$  is sceptically accepted if it is a member of every preferred extension.

The concepts of credulous and sceptical acceptance motivate the decision problems of Table 1 that have been considered in [9,15].

**Table 1.** Decision Problems in Argumentation Frameworks

Decision Problem	Instance	Question	Complexity
CA	$\mathcal{H}(\mathcal{X}, \mathcal{A}), x \in \mathcal{X}$	Is $x$ credulously accepted in $\mathcal{H}$ ?	NP-complete [9]
SA	$\mathcal{H}(\mathcal{X}, \mathcal{A}), x \in \mathcal{X}$	Is $x$ sceptically accepted in $\mathcal{H}$ ?	$\Pi_2^P$ complete [15]

We consider a number of decision problems relating to properties of ideal extensions in argumentation frameworks as described in Table 2.

**Table 2.** Decision questions for Ideal Semantics

Problem Name	Instance	Question
IE	$\mathcal{H}(\mathcal{X}, \mathcal{A}); S \subseteq \mathcal{X}$	Is $S$ an ideal extension?
IA	$\mathcal{H}(\mathcal{X}, \mathcal{A}); x \in \mathcal{X}$	Is $x$ in the maximal ideal extension?
MIE $_\emptyset$	$\mathcal{H}(\mathcal{X}, \mathcal{A})$	Is the maximal ideal extension empty?
MIE	$\mathcal{H}(\mathcal{X}, \mathcal{A}); S \subseteq \mathcal{X}$	Is $S$ the maximal ideal extension?
CS	$\mathcal{H}(\mathcal{X}, \mathcal{A})$	Is $\mathcal{H}(\mathcal{X}, \mathcal{A})$ cohesive?

We also consider so-called *function problems* where the aim is not simply to verify that a given set has a specific property but to *construct* an example. In particular we examine the problem FMIE in which given an AF it is required to return its maximal ideal extension.

We recall that  $D^P$  is the class of decision problems,  $L$ , whose positive instances are characterised as those belonging to  $L_1 \cap L_2$  where  $L_1 \in \text{NP}$  and  $L_2 \in \text{co-NP}$ . The problem SAT-UNSAT whose instances are pairs of 3-CNF formulae  $\langle \Phi_1, \Phi_2 \rangle$  accepted if  $\Phi_1$  is satisfiable and  $\Phi_2$  is unsatisfiable has been shown to be complete for this class [18, p. 413]. This class can be interpreted as those decision problems that may be solved by a (deterministic) polynomial time algorithm which is allowed to make at most two calls upon an NP *oracle*. More generally, the complexity classes  $P^{NP}$  and  $FP^{NP}$  consist of those decision problems (respectively function problems) that can be solved by a (deterministic) polynomial time algorithm provided with access to an NP oracle (calls upon which take a single step so that only polynomially many invocations of this oracle are allowed).<sup>2</sup> An important (presumed) subset of  $P^{NP}$  and its associated function class is defined by distinguishing whether oracle calls are *adaptive* – i.e. the exact formulation of the next oracle query may be dependent on the answers received to previous questions – or whether such queries are *non-adaptive*, i.e. the form of the questions to be put to the oracle is predetermined allowing all of these to be performed in parallel. The latter class, which we denote

<sup>2</sup>We refer the reader to e.g. [18, pp. 415–423] for further background concerning these classes.

by  $P_{||}^{NP}$ , has been considered in Wagner [21,22], Jenner and Toran [16]. Under the standard complexity-theoretic assumptions, it is conjectured that,

$$P \subset \left\{ \begin{array}{l} NP \\ co-NP \end{array} \right\} \subset D^P \subset P_{||}^{NP} \subset P^{NP} \subset \left\{ \begin{array}{l} \Sigma_2^P \\ \Pi_2^P \end{array} \right\}$$

We prove the following complexity classifications.

- a. IE is co-NP-complete.
- b. IA is co-NP-hard via  $\leq_m^p$ -reducibility.
- c. MIE $_\emptyset$  is NP-hard via  $\leq_m^p$ -reducibility.
- d. MIE is  $D^P$ -hard via  $\leq_m^p$ -reducibility.
- e. CS is  $\Sigma_2^P$ -complete.
- f. FMIE is  $FP_{||}^{NP}$ -complete.
- g. Problems (a)–(f) are polynomial time solvable for *bipartite* frameworks.<sup>3</sup>
- h. Problems (b)–(d) are  $P_{||}^{NP}$ -complete via *randomized* reductions.

## 2. Characteristic properties of ideal extensions

A number of our results exploit the characterisation of ideal extensions in terms of credulous acceptability given in Lemma 1. We also present, in Lemma 2, a necessary and sufficient condition for a given *argument* to be a member of the maximal ideal extension.

**Lemma 1** *Let  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  be an argumentation framework and  $S \subseteq \mathcal{X}$ . The set of arguments  $S$  defines an ideal extension of  $\mathcal{H}$  if and only if both of the conditions below are satisfied:*

- I1. *S is an admissible set of arguments in  $\mathcal{H}$ .*
- I2. *For every argument  $p \in S^-$ , there is no admissible set of  $\mathcal{H}$  that contains p, i.e.  $\forall p \in S^- \neg CA(\mathcal{H}, p)$ .*

**Lemma 2** *Let  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  be an AF and let  $\mathcal{M} \subseteq \mathcal{X}$  be the maximal ideal extension of  $\mathcal{H}(\mathcal{X}, \mathcal{A})$ . Then  $x \in \mathcal{X}$  is a member of  $\mathcal{M}$  if and only if both of the conditions below are satisfied:*

- M1. *No attacker of x is credulously accepted, i.e.  $\forall y \in \{x\}^- \neg CA(\mathcal{H}, y)$ .*
- M2. *For each attacker y of x, at least one attacker z of y is in  $\mathcal{M}$ , i.e.  $\forall y \in \{x\}^- : \{y\}^- \cap \mathcal{M} \neq \emptyset$ .*

## 3. Decision questions in ideal argumentation

**Theorem 1** *IE is co-NP-complete.*

---

<sup>3</sup>An AF,  $\langle \mathcal{X}, \mathcal{A} \rangle$  is *bipartite* if  $\mathcal{X}$  can be partitioned into two sets,  $\mathcal{Y}$  and  $\mathcal{Z}$  both of which are conflict-free in  $\langle \mathcal{X}, \mathcal{A} \rangle$ .

**Proof:** (Outline) Given an instance  $\langle \mathcal{H}(\mathcal{X}, \mathcal{A}), S \rangle$  of IE we can decide in co-NP if the instance should be accepted by checking whether  $S$  is admissible and that no admissible set contains an attacker of  $S$ . Correctness follows from Lemma 1.

To prove IE is co-NP-hard we reduce from CNF-UNSAT (without loss of generality, restricted to instances which are 3-CNF). Given a 3-CNF formula

$$\Phi(z_1, \dots, z_n) = \bigwedge_{i=1}^m C_i = \bigwedge_{i=1}^m (z_{i,1} \vee z_{i,2} \vee z_{i,3})$$

as an instance of UNSAT we form an instance  $\langle \mathcal{F}_\Phi, S \rangle$  of IE as follows. First form the AF  $\mathcal{H}_\Phi(\mathcal{X}, \mathcal{A})$  with

$$\begin{aligned} \mathcal{X} &= \{\Phi, C_1, \dots, C_m\} \cup \{z_i, \neg z_i : 1 \leq i \leq n\} \\ \mathcal{A} &= \{\langle C_j, \Phi \rangle : 1 \leq j \leq m\} \cup \{\langle z_i, \neg z_i \rangle, \langle \neg z_i, z_i \rangle : 1 \leq i \leq n\} \cup \\ &\quad \{\langle z_i, C_j \rangle : z_i \text{ occurs in } C_j\} \cup \{\langle \neg z_i, C_j \rangle : \neg z_i \text{ occurs in } C_j\} \end{aligned}$$

Via [9, Thm. 5.1, p. 227], the argument  $\Phi$  is credulously accepted if and only if the CNF,  $\Phi(Z_n)$  is satisfiable, i.e.  $\Phi$  is *not* credulously accepted if and only if  $\Phi(Z_n)$  is unsatisfiable. The AF,  $\mathcal{F}_\Phi$ , is formed from  $\mathcal{H}_\Phi$  by adding an argument  $\Psi$  together with attacks

$$\{\langle \Psi, z_i \rangle, \langle \Psi, \neg z_i \rangle : 1 \leq i \leq n\} \cup \{\langle \Phi, \Psi \rangle, \langle \Psi, \Phi \rangle\}$$

The instance of IE is completed by setting  $S = \{\Psi\}$ .

It can be shown that  $\langle \mathcal{F}_\Phi, \{\Psi\} \rangle$  is accepted as an instance of IE if and only if  $\Phi$  is unsatisfiable. We omit the detailed analysis.  $\square$

### Corollary 1

- a. IA is co-NP-hard.
- b. MIE $_\emptyset$  is NP-hard
- c. MIE is D $^p$ -hard.

**Theorem 2** CS is  $\Sigma_2^p$ -complete.

**Proof:** Omitted.  $\square$

**Corollary 2** The property of coherence is neither necessary nor sufficient for an AF to be cohesive.

**Proof:** The proof of Thm. 2 uses the AF from [15, p. 198]: its set of sceptically accepted arguments is empty if and only if the same system fails to be coherent. Furthermore if this system is coherent, then it contains exactly one sceptically accepted argument and this argument fails to define an admissible set. Hence the system is cohesive if and only if it is *not* coherent.  $\square$

#### 4. Finding the Maximal Ideal Extension

**Theorem 3** FMIE is  $\text{FP}_{\parallel}^{\text{NP}}$ -complete.

**Proof:** (Outline) We consider only the membership argument. Let  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  be an AF and consider the following partition of  $\mathcal{X}$ .

$$\begin{aligned}\mathcal{X}_{\text{OUT}} &= \{x \in \mathcal{X} : \neg \text{CA}(\mathcal{H}, x)\} \\ \mathcal{X}_{\text{PSA}} &= \{x \in \mathcal{X} : \{x\}^- \cup \{x\}^+ \subseteq \mathcal{X}_{\text{OUT}}\} \setminus \mathcal{X}_{\text{OUT}} \\ \mathcal{X}_{\text{CA}} &= \{x \in \mathcal{X} : \text{CA}(\mathcal{H}, x)\} \setminus \mathcal{X}_{\text{PSA}}\end{aligned}$$

With this partition we may construct a bipartite framework –  $\mathcal{B}(\mathcal{X}_{\text{PSA}}, \mathcal{X}_{\text{OUT}}, \mathcal{F})$  – in which the set of attacks,  $\mathcal{F}$ , is

$$\mathcal{F} =_{\text{def}} \mathcal{A} \setminus \{\langle y, z \rangle : y \in \mathcal{X}_{\text{CA}} \cup \mathcal{X}_{\text{OUT}} \text{ and } z \in \mathcal{X}_{\text{CA}} \cup \mathcal{X}_{\text{OUT}}\}$$

The  $\text{FP}_{\parallel}^{\text{NP}}$  upper bound now follows by noting that the partition  $(\mathcal{X}_{\text{PSA}}, \mathcal{X}_{\text{CA}}, \mathcal{X}_{\text{OUT}})$  can be constructed in  $\text{FP}_{\parallel}^{\text{NP}}$  and the maximal ideal extension of  $\mathcal{H}$  is the maximal admissible subset of  $\mathcal{X}_{\text{PSA}}$  in the bipartite graph  $\mathcal{B}(\mathcal{X}_{\text{PSA}}, \mathcal{X}_{\text{OUT}}, \mathcal{F})$ . Using the algorithm of [14, Thm. 6(a)] this set can be found in polynomial time and thus  $\text{FMIE} \in \text{FP}_{\parallel}^{\text{NP}}$  as claimed.  $\square$

**Corollary 3**  $\mathcal{H}(\mathcal{X}, \mathcal{A})$  is cohesive if every argument  $x \in \mathcal{X}$  is credulously accepted.

The properties of bipartite AFs exploited in Thm. 3 are also central to the following result.

**Theorem 4** If  $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$  is a bipartite AF then  $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$  is cohesive.

**Proof:** (Outline) The set  $\mathcal{M}$  of sceptically accepted arguments in  $\mathcal{B}$  is the union of  $\mathcal{Y}_{\text{SA}}$  (sceptically accepted arguments in  $\mathcal{Y}$ ) with  $\mathcal{Z}_{\text{SA}}$  (sceptically accepted arguments in  $\mathcal{Z}$ ). By considering the maximal admissible subsets  $S_{\mathcal{Y}}$  of  $\mathcal{Y}$  and  $S_{\mathcal{Z}}$  of  $\mathcal{Z}$  in the AF  $\mathcal{B}$ , it can be shown that  $\mathcal{M}$  is admissible.  $\square$

**Corollary 4** Let  $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$  be a bipartite AF. The maximal ideal extension of  $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$  may be constructed in polynomial time.

**Proof:** From Thm. 4 the maximal ideal extension of  $\mathcal{B}$  corresponds with the set of all sceptically accepted arguments of  $\mathcal{B}$ . Applying the methods described in [14, Thm. 6] this set can be identified in polynomial time.  $\square$

We note that as a consequence of Thm. 4 and Corollary 4 in the case of bipartite AFs, the decision problems IE, MIE and IA are all in P and CS is trivial. Combining the results we obtain the picture of the relative complexities of checking whether an argument is credulously/ideally/sceptically acceptable shown in Table 3.

Similarly, Table 4 considers checking whether a given set of arguments collectively satisfies the requirements of a given semantics or is a maximal such set.

In total the classifications given by these tables reinforces the the case that deciding credulous admissibility semantics is easier than ideal semantics which, in turn, is easier than sceptical admissibility semantics.

**Table 3.** Relative Complexity of Testing Acceptability.

Acceptability Semantics	Lower bound	Upper Bound
CA	NP-hard	NP
IA	co-NP-hard	$P_{  }^{NP}$
SA	$\Pi_2^P$ -hard	$\Pi_2^P$

**Table 4.** Deciding set and maximality properties

Semantics	Problem	Lower bound	Upper Bound
Credulous	ADM	P	P
Ideal	IE	co-NP-hard	co-NP
Credulous	PREF-EXT	co-NP-hard	co-NP
Ideal	MIE	$D^P$ -hard	$P_{  }^{NP}$
Ideal	FMIE	$FP_{  }^{NP}$ -hard	$FP_{  }^{NP}$

## 5. Reducing the complexity gaps

The results given exhibit a gap between lower and upper bounds. We now briefly present evidence that  $IA \not\in \text{co-NP}$ . This relies on structural complexity results derived in Chang *et al.* [4,5] as a consequence of which we obtain:

### Theorem 5

- a. If IA is NP-hard then IA is  $P_{||}^{NP}$ -complete.
- b. If  $IA \in \text{co-NP}$  then MIE is  $D^P$ -complete.
- c. If  $IA \in \text{co-NP}$  then  $MIE_\emptyset$  is NP-complete.

**Proof:** Omitted. □

We may interpret Thm. 5 as focusing the issue of obtaining exact classifications in terms of IA. In fact, there is strong evidence that  $IA \not\in \text{co-NP}$  and, using one suite of techniques is more likely to be complete within  $P_{||}^{NP}$ . Our formal justification of these claims rests on a number of technical analyses using results of Chang *et al.* [5], which in turn develop ideas of [1,2,19]. Two key concepts in our further analyses of IA are,

- a. The so-called *Unique Satisfiability* problem (USAT).
- b. Randomized reductions between languages.

### Unique Satisfiability (USAT)

**Instance:** CNF formula  $\Phi(X_n)$  with propositional variables  $\langle x_1, \dots, x_n \rangle$ .

**Question:** Does  $\Phi(X_n)$  have *exactly one* satisfying instantiation?

Determining the exact complexity of USAT remains an open problem. It is known that  $USAT \in D^P$  and while Blass and Gurevich [2] show it to be co-NP-hard<sup>4</sup>, USAT has only been shown to be complete for  $D^P$  using a *randomized* reduction technique of Valiant and Vazirani [19].

---

<sup>4</sup>The reader should note that [17, p. 93] has a typographical slip whereby Blass and Gurevich's result is described as proving USAT to be NP-hard.

**Definition 2** Let  $L_1$  and  $L_2$  be languages and  $\delta \in [0, 1]$ . We say that  $L_1$  randomly reduces to  $L_2$  (denoted  $L_1 \leq_m^{rp} L_2$ ) with probability  $\delta$  if there is a polynomial time computable function,  $f$ , and polynomial bound  $q$  with  $f$  mapping pairs  $\langle x, z \rangle - x$  an instance of  $L_1$  and  $z$  an element of  $\langle 0, 1 \rangle^{q(|x|)}$  – to instances,  $y$ , of  $L_2$ , such that for  $z$  drawn uniformly at random from  $\langle 0, 1 \rangle^{q(|x|)}$

$$\begin{aligned} x \in L_1 &\Rightarrow \text{Prob}[f(x, z) \in L_2] \geq \delta \\ x \notin L_1 &\Rightarrow \text{Prob}[f(x, z) \notin L_2] = 1 \end{aligned}$$

We have the following properties of USAT and randomized reductions:

**Fact 1** SAT  $\leq_m^{rp}$  USAT with probability  $1/(4n)$ . ([19, Lemma 2.1, p. 88])

Although the “success probability”  $1/(4n)$  approaches 0 as  $n$  increases, in suitable cases [5, Fact 1, p. 361] show that this may be amplified to  $1 - 2^{-n}$ : the problems IA and MIE $_\emptyset$  both satisfy the criteria for such stronger randomized reductions to be built. We refer the reader to [13] for details.

A relationship between unique satisfiability (USAT) and ideal acceptance (IA) is established in the following theorem. Notice that the reduction we describe is *deterministic*, i.e. not randomized.

**Theorem 6** USAT  $\leq_m^p$  IA.

**Proof:** Given an instance  $\Phi(Z_n)$  of USAT construct an AF,  $\mathcal{K}_\Phi(\mathcal{X}, \mathcal{A})$  as follows. First form the system  $\mathcal{F}_\Phi$  described in Thm. 1, but without the attack  $\langle \Psi, \Phi \rangle$  contained in this and with attacks  $\langle C_j, C_j \rangle$  for each clause of  $\Phi$ .<sup>5</sup> We then add a further  $n + 1$  arguments,  $\{y_1, \dots, y_n, x\}$  and attacks

$$\{\langle z_i, y_i \rangle, \langle \neg z_i, y_i \rangle : 1 \leq i \leq n\} \cup \{\langle y_i, x \rangle : 1 \leq i \leq n\}$$

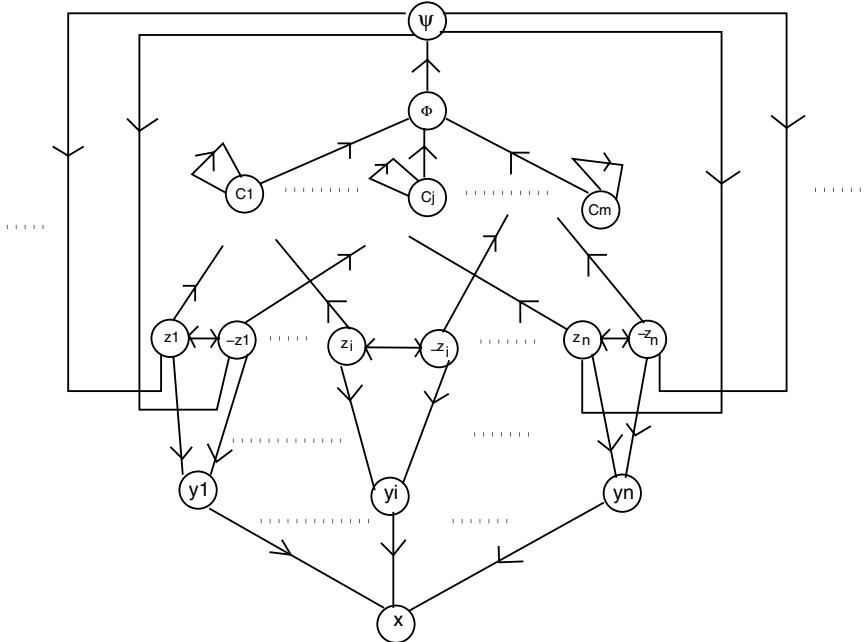
The instance of IA is  $\langle \mathcal{K}_\Phi(\mathcal{X}, \mathcal{A}), x \rangle$  and the resulting AF is illustrated in Fig. 1.

We now claim that  $\Phi(Z_n)$  has a unique satisfying instantiation if and only if  $x$  is a member of  $\mathcal{M}_\mathcal{K}$  the maximal ideal extension of  $\mathcal{K}_\Phi(\mathcal{X}, \mathcal{A})$ .

Suppose first that  $\Phi(Z_n)$  does *not* have a unique satisfying instantiation. If  $\Phi$  is unsatisfiable – i.e. the number of satisfying assignments is zero – then no argument of  $\mathcal{K}_\Phi$  is credulously accepted, thus  $x \notin \mathcal{M}_\mathcal{K}$ . There remains the possibility that  $\Phi(Z_n)$  has two or more satisfying assignments. Suppose  $\alpha = \langle a_1, a_2, \dots, a_n \rangle$  and  $\beta = \langle b_1, b_2, \dots, b_n \rangle$  are such that  $\Phi(\alpha) = \Phi(\beta) = \top$  and  $\alpha \neq \beta$ . Without loss of generality, we may assume that  $a_1 \neq b_1$  (since  $\alpha \neq \beta$  there must be at least one variable of  $Z_n$  that is assigned differing values in each). In this case *both*  $z_1$  and  $\neg z_1$  are credulously accepted so that neither can belong to  $\mathcal{M}_\mathcal{K}$ : from Lemma 2 condition (M1) gives  $z_1 \notin \mathcal{M}_\mathcal{K}$  (since  $\neg z_1$  is credulously

---

<sup>5</sup>We make these arguments self-attacking purely for ease of presentation: the required effect – that no argument  $C_j$  is ever credulously accepted – can be achieved without self-attacks simply by adding two arguments  $d_j$  and  $e_j$  for each clause together with attacks  $\{\langle C_j, d_j \rangle, \langle d_j, e_j \rangle, \langle e_j, C_j \rangle\}$ .

Figure 1. The Argumentation Framework  $\mathcal{K}_\Phi$ 

accepted) and  $\neg z_1 \notin \mathcal{M}_K$  (since  $z_1$  is credulously accepted). It now follows that  $x \notin \mathcal{M}_K$  via (M2) of Lemma 2: neither attacker of  $y_1$ , an argument which attacks  $x$ , belongs to  $\mathcal{M}_K$ . We deduce that if  $\Phi(Z_n)$  is not a positive instance of USAT then  $\langle \mathcal{K}_\Phi, x \rangle$  is not a positive instance of IA.

On the other hand suppose that  $\alpha = \langle a_1, a_2, \dots, a_n \rangle$  defines the unique satisfying instantiation of  $\Phi(Z_n)$ . Consider the following subset of  $\mathcal{X}$ :

$$\mathcal{M} = \bigcup_{i : a_i = \top} \{z_i\} \cup \bigcup_{i : a_i = \perp} \{\neg z_i\} \cup \{\Phi, x\}$$

Certainly  $\mathcal{M}$  is admissible: since  $\alpha$  satisfies  $\Phi(Z_n)$  each  $C_j$  and  $y$  is attacked by some  $z$  or  $\neg z$  in  $\mathcal{M}$  and thus all of the attacks on  $\Phi$  and  $x$  are counterattacked. Similarly  $\Phi$  defends arguments against the attacks by  $\Psi$ . It is also the case, however, that no admissible set of  $\mathcal{K}_\Phi$  contains an attacker of  $\mathcal{M}$ . No admissible set can contain  $C_j$  (since these arguments are self-attacking),  $\Psi$  (since the only defenders of the attack by  $\Phi$  are  $C_j$  arguments) or  $y_k$  ( $1 \leq k \leq n$ ) (since these require  $\Psi$  as a defence against  $\{z_k, \neg z_k\}$ ). Furthermore for  $z_i \in \mathcal{M}$  an admissible set containing  $\neg z_i$  would only be possible if there were a satisfying assignment of  $\Phi$  under which  $\neg z_i = \top$ : this would contradict the assumption the  $\Phi$  had exactly one satisfying instantiation.

We deduce that  $\Phi(Z_n)$  has a unique satisfying instantiation if and only if  $x$  is in the maximal ideal extension of  $\mathcal{K}_\Phi(\mathcal{X}, \mathcal{A})$ .  $\square$

**Corollary 5** USAT  $\leq_m^p \neg \text{MIE}_\emptyset$

**Proof:** The AF  $\mathcal{K}_\Phi$  of Thm. 6 has a *non-empty* maximal ideal extension,  $\mathcal{M}_\mathcal{K}$ , if and only if  $x \in \mathcal{M}_\mathcal{K}$ .  $\square$

Combining Thms. 5 and 6 with Fact 1 and [5, Fact 1, p. 361] gives the following corollaries.

**Corollary 6** *Each of the decision problems in  $\{\text{IA}, \text{MIE}, \text{MIE}_\emptyset\}$  is complete for  $P_{||}^{\text{NP}}$  under  $\leq_m^{rp}$  reductions with probability  $1 - 2^{-n}$ .*

To conclude we observe that although  $\text{USAT} \leq_m^p \text{IA}$  it is unlikely to be the case that these decision problems have equivalent complexity, i.e. that  $\text{IA} \leq_m^p \text{USAT}$ .

**Corollary 7** *If  $\text{IA} \leq_m^p \text{USAT}$  (note deterministic reduction) then the Polynomial Hierarchy (PH) collapses to  $\Sigma_3^p$ , i.e.*

$$\text{IA} \leq_m^p \text{USAT} \Rightarrow \bigcup_{k \geq 3} \Sigma_k^p \cup \bigcup_{k \geq 3} \Pi_k^p \subseteq \Sigma_3^p$$

Now, noting that  $\leq_m^p$  reductions can be interpreted as “ $\leq_m^{rp}$  reductions with probability 1”, we can reconsider the lower bounds of Tables 3 and 4 using hardness via  $\leq_m^{rp}$  reductions (with “high” probability) instead of hardness via deterministic  $\leq_m^p$  reducibility, as shown in Table 5.

**Table 5.** Complexity of ideal semantics relative to randomized reductions

Decision Problem	Complexity	$\leq_m^{rp}$ probability
CA	NP-complete	1
IA	$P_{  }^{\text{NP}}$ -complete	$1 - 2^{-n}$
SA	$\Pi_2^p$ -complete	1
ADM	P	—
IE	co-NP-complete	1
PREF-EXT	co-NP-complete	1
PREF-EXT $_\emptyset$	co-NP-complete	1
MIE	$P_{  }^{\text{NP}}$ -complete	$1 - 2^{-n}$
MIE $_\emptyset$	$P_{  }^{\text{NP}}$ -complete	$1 - 2^{-n}$

## 6. Conclusions and Further Work

We have considered the computational complexity of decision and search problems arising in the Ideal semantics for abstract argumentation frameworks introduced in [11,12]. It has been shown that all but one of these<sup>6</sup> can be resolved within  $P_{||}^{\text{NP}}$  or its functional analogue  $\text{FP}_{||}^{\text{NP}}$ : classes believed to lie strictly below the second level of the polynomial hierarchy. We have, in addition, presented compelling evidence that deciding if an argument is acceptable under the ideal semantics, if

<sup>6</sup>The exception being the problem CS.

a set of arguments defines the maximal ideal extension, and if the maximal ideal extension is empty, are not contained within any complexity class falling strictly within  $P_{\parallel}^{NP}$ : all of these problems being  $P_{\parallel}^{NP}$ -hard with respect to  $\leq_m^{rp}$  reductions of probability  $1 - 2^{-n}$ . Although this complexity class compares unfavourably with the NP and co-NP-complete status of related questions under the credulous preferred semantics of [10], it represents an improvement on the  $\Pi_2^p$ -completeness level of similar issues within the sceptical preferred semantics.

Given that a precondition of ideal acceptance is that the argument is sceptically accepted this reduction in complexity may appear surprising. The apparent discrepancy is, however, accounted for by examining the second condition that a *set* of arguments must satisfy in order to form an ideal extension: as well as being sceptically accepted, the set must be admissible. This condition plays a significant role in the complexity shift. An important reason why testing sceptical acceptance of a given argument  $x$  fails to belong to co-NP (assuming co-NP  $\neq \Pi_2^p$ ) is that the condition “no attacker of  $x$  is credulously accepted” while *necessary* for sceptical acceptance of  $x$  is not *sufficient*: a fact which seems first to have been observed by Vreeswijk and Prakken [20] in their analysis of sound and complete proof procedures for credulous acceptance. Although this condition *is* sufficient in *coherent* frameworks, deciding if  $\mathcal{H}$  is coherent is already  $\Pi_2^p$ -complete [15]. In contrast, as demonstrated in the characterisation of ideal extensions given in Lemma 1, an *admissible* set,  $S$ , is also sceptically accepted if and only if no argument in  $S^-$  is credulously accepted: we thus have a condition which can be tested in co-NP.

The reason why *finding* the maximal ideal extension (and consequently decision questions predicated on its properties, e.g. cardinality, membership, etc.) can be performed more efficiently than testing sceptical acceptance stems from the fact this set can be readily computed given the *bipartite* framework,  $\mathcal{B}(\mathcal{X}_{PSA}, \mathcal{X}_{OUT}, \mathcal{F})$  associated with  $\mathcal{H}(\mathcal{X}, \mathcal{A})$ . Construction of this framework only requires determining the set,  $\mathcal{X}_{OUT}$ , of arguments which are not credulously accepted, so that *explicit* consideration of sceptical acceptance is never required.

This paper has focussed on the graph-theoretic abstract argumentation framework model from [10]. A natural continuation, and the subject of current work, is to consider the divers instantiations of *assumption-based* argumentation frameworks (ABF) [3]. Complexity-theoretic analyses of this model with respect to credulous and sceptical semantics have been presented in a series of papers by Dimopoulos, Nebel, and Toni [6,7,8]. In these, the computational complexity of specific decision questions is shown to be linked with that of deciding  $\Delta \models \varphi$  where  $\Delta$  is a given collection of formulae and  $\models$  is a derivability relation whose precise semantics are dependent on the logical theory described by the ABF, e.g. [3] describe how ABFs may be formulated to capture a variety of non-classical logics. While one might reasonably expect a number of the techniques described above to translate to ABF instantiations, there are non-trivial issues for cases where  $\Delta \models \varphi$  is NP-complete, e.g. the default logic instantiation of ABFs. A significant problem in such cases concerns the mechanisms explored in Section 5 in order to amplify the co-NP-hardness (via  $\leq_m^p$  reductions) of IA to  $P_{\parallel}^{NP}$ . For example, the reduction from USAT and the technical machinery of [4,5] whether a “natural” analogue of USAT for the second level of the polynomial hierarchy can be formulated is far from clear.

## References

- [1] L. M. Adleman and K. Manders. Reducibility, randomness and intractability. In *Proc. 9th ACM Symposium on Theory of Computing*, pages 151–163, 1979.
- [2] A. Blass and Y. Gurevich. On the unique satisfiability problem. *Information and Control*, 55:80–82, 1982.
- [3] A. Bondarenko, P.M. Dung, R.A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93:63–101, 1997.
- [4] R. Chang and J. Kadin. On computing Boolean connectives of characteristic functions. *Math. Syst. Theory*, 28:173–198, 1995.
- [5] R. Chang, J. Kadin, and P. Rohatgi. On unique satisfiability and the threshold behavior of randomised reductions. *Jnl. of Comp. and Syst. Sci.*, pages 359–373, 1995.
- [6] Y. Dimopoulos, B. Nebel, and F. Toni. Preferred arguments are harder to compute than stable extensions. In D. Thomas, editor, *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99-Vol1)*, pages 36–43, San Francisco, 1999. Morgan Kaufmann Publishers.
- [7] Y. Dimopoulos, B. Nebel, and F. Toni. Finding admissible and preferred arguments can be very hard. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *KR2000: Principles of Knowledge Representation and Reasoning*, pages 53–61, San Francisco, 2000. Morgan Kaufmann.
- [8] Y. Dimopoulos, B. Nebel, and F. Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141:55–78, 2002.
- [9] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science*, 170:209–244, 1996.
- [10] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, 77:321–357, 1995.
- [11] P. M. Dung, P. Mancarella, and F. Toni. A dialectical procedure for sceptical assumption-based argumentation. In P. E. Dunne and T. J. M. Bench-Capon, editors, *Proc. 1st Int. Conf. on Computational Models of Argument*, volume 144 of *FAIA*, pages 145–156. IOS Press, 2006.
- [12] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171:642–674, 2007.
- [13] P. E. Dunne. The computational complexity of ideal semantics I: abstract argumentation frameworks. Technical Report ULCS-07-008, Dept. of Comp. Sci., Univ. of Liverpool, June 2007.
- [14] P. E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artificial Intelligence*, 171:701–729, 2007.
- [15] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141:187–203, 2002.
- [16] B. Jenner and J. Toran. Computing functions with parallel queries to NP. *Theoretical Computer Science*, 141:175–193, 1995.
- [17] D. S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science. Volume A: Algorithms and Complexity*, pages 67–161. Elsevier Science, 1998.
- [18] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.
- [19] L. G. Valiant and V. V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science*, 47:85–93, 1986.
- [20] G. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proceedings of JELIA'2000, The 7th European Workshop on Logic for Artificial Intelligence.*, pages 224–238, Berlin, 2000. Springer LNAI 1919, Springer Verlag.
- [21] K. Wagner. Bounded query computations. In *Proc. 3rd Conf. on Structure in Complexity Theory*, pages 260–277, 1988.
- [22] K. Wagner. Bounded query classes. *SIAM Jnl. Comput.*, 19:833–846, 1990.

# Focused search for arguments from propositional knowledge

Vasiliki EFSTATHIOU<sup>a</sup> and Anthony HUNTER<sup>a</sup>

<sup>a</sup> *Department of Computer Science,*

*University College London,*

*Gower Street, London WC1E 6BT, UK*

{v.efstathiou, a.hunter}@cs.ucl.ac.uk

**Abstract** Classical propositional logic is an appealing option for modelling argumentation but the computational viability of generating an argument is an issue. Here we propose ameliorating this problem by harnessing the notion of a connection graph to reduce the search space when seeking all the arguments for a claim from a knowledgebase. For a set of clauses, a connection graph is a graph where each node is a clause and each arc denotes that there exist complementary disjuncts in the pair of nodes. For a set of formulae in conjunctive normal form, we use the notion of the connection graph for the set of clauses obtained from the conjuncts in the formulae. When seeking arguments for a claim, we can focus our search on a particular subgraph of the connection graph that we call the focal graph. Locating this subgraph is relatively inexpensive in terms of computational cost. In addition, using (as the search space) the formulae of the initial knowledgebase, whose conjuncts relate to this subgraph, can substantially reduce the cost of looking for arguments. We provide a theoretical framework and algorithms for this proposal, together with some theoretical results and some preliminary experimental results to indicate the potential of the approach.

## 1. Introduction

Argumentation is a vital aspect of intelligent behaviour by humans. Consider diverse professionals such as politicians, journalists, clinicians, scientists, and administrators, who all need to collate and analyse information looking for pros and cons for consequences of importance when attempting to understand problems and make decisions.

There are a number of proposals for logic-based formalisations of argumentation (for reviews see [7,18,4]). These proposals allow for the representation of arguments for and against some claim, and for counterargument relationships between arguments. In a number of key examples of argumentation systems, an argument is a pair where the first item in the pair is a minimal consistent set of formulae that proves the second item which is a formula (see for example [2,11,3,1,12]). Proof procedures and algorithms have been developed for finding preferred arguments from a knowledgebase using defeasible logic and following for example Dung's preferred semantics (see for example [5,20,17,14,6,8,9]). However, these techniques and analyses do not offer any ways of ameliorating the computational complexity inherent in finding arguments for classical logic.

Suppose we use an automated theorem prover (an ATP). If we seek arguments for a particular claim  $\alpha$ , we need to post queries to the ATP to ensure that a particular set of premises entails  $\alpha$ , that the set of premises is minimal for this, and that it is consistent. So finding arguments for a claim  $\alpha$  involves considering subsets  $\Phi$  of  $\Delta$  and testing them with the ATP to ascertain whether  $\Phi \vdash \alpha$  and  $\Phi \not\vdash \perp$  hold. For  $\Phi \subseteq \Delta$ , and a formula  $\alpha$ , let  $\Phi?\alpha$  denote a call (a query) to an ATP. If  $\Phi$  classically entails  $\alpha$ , then we get the answer  $\Phi \vdash \alpha$ , otherwise we get the answer  $\Phi \not\vdash \alpha$ . In this way, we do not give the whole of  $\Delta$  to the ATP. Rather we call it with particular subsets of  $\Delta$ . So for example, if we want to know if  $\langle \Phi, \alpha \rangle$  is an argument, then we have a series of calls  $\Phi?\alpha$ ,  $\Phi?\perp$ ,  $\Phi \setminus \{\phi_1\}?\alpha, \dots, \Phi \setminus \{\phi_k\}?\alpha$ , where  $\Phi = \{\phi_1, \dots, \phi_k\}$ . So the first call is to ensure that  $\Phi \vdash \alpha$ , the second call is to ensure that  $\Phi \not\vdash \perp$ , the remaining calls are to ensure that there is no subset  $\Phi'$  of  $\Phi$  such that  $\Phi' \vdash \alpha$ . This then raises the question of which subsets  $\Phi$  of  $\Delta$  to investigate when we are searching for an argument for  $\alpha$ . Moreover, if we want to find all arguments for a claim in  $\Delta$ , in the worst case we need to consider all subsets of  $\Delta$ .

It is with these issues in mind that we explore an alternative way of finding all the arguments from a knowledgebase  $\Delta$  for a claim  $\alpha$ . Our approach is to adapt the idea of connection graphs to enable us to find arguments. A connection graph [15,16] is a graph where a clause is represented by a node and an arc  $(\phi, \psi)$  denotes that there is a disjunct in  $\phi$  with its complement being a disjunct in  $\psi$ . In previous work [10], we have proposed a framework for using connection graphs for finding arguments. However, in that paper we restricted consideration to knowledgebases of clauses and claims to being literals. Furthermore, in that paper we did not give the algorithms for constructing the connection graphs, but rather we focused on algorithms for searching through graph structures for supports for arguments. Finally, we did not provide a systematic empirical evaluation of the algorithms for constructing graphs. We address these three shortcomings in this paper.

So, in this paper we propose algorithms for isolating a particular subset of the knowledgebase which essentially contains all the subsets of the knowledgebase that can be supports for arguments for a given claim. Initially we restrict the language used to a language of (disjunctive) clauses and we describe how arranging a clause knowledgebase in a connection graph structure can help focusing on a subgraph of the initial one that corresponds to the subset of a knowledgebase connected to a given clause. Furthermore, we illustrate how this approach can be generalised for a language of propositional formulae in conjunctive normal form where the clauses of interest in this case are the conjuncts of the negated claim for an argument. We describe how this method can be efficient regarding the search space reduction and hence, the computational cost of finding arguments. Finally, we present some experimental results illustrating how the software implementation of these algorithms performs.

## 2. Preliminaries

In this section, we review an existing proposal for logic-based argumentation [3] together with a recent proposal for using connection graphs in argumentation [10].

We consider a classical propositional language with classical deduction denoted by the symbol  $\vdash$ . We use  $\Delta, \Phi, \dots$  to denote sets of formulae,  $\phi, \psi, \dots$  to denote formulae

and  $a, b, c \dots$  to denote the propositional letters each formula consists of. For the following definitions we first assume a knowledgebase  $\Delta$  (a finite set of formulae) and we use this  $\Delta$  throughout. Furthermore we assume that each of the formulae of  $\Delta$  is in conjunctive normal form (i.e. a conjunction of one or more disjunctive clauses). We use  $\Delta$  as a large repository of information from which arguments can be constructed for and against arbitrary claims. The framework adopts a common intuitive notion of an argument. Essentially an argument is a set of relevant formulae from  $\Delta$  that can be used to minimally and consistently entail a claim together with that claim. In this paper each claim is represented by a formula represented in conjunctive normal form.

**Definition 1.** An argument is a pair  $\langle \Phi, \phi \rangle$  such that (1)  $\Phi \subseteq \Delta$ , (2)  $\Phi \not\vdash \perp$ , (3)  $\Phi \vdash \phi$  and (4) there is no  $\Phi' \subset \Phi$  s.t.  $\Phi' \vdash \phi$ .

**Example 1.** Let  $\Delta = \{\neg a, (\neg a \vee b) \wedge c, (d \vee e) \wedge f, \neg b \wedge d, (\neg f \vee g) \wedge (a \vee \neg e), \neg e \vee e, \neg k \vee m, \neg m\}$ . Some arguments are :  $\langle \{\neg a, (d \vee e) \wedge f\}, \neg a \wedge (d \vee e) \rangle$ ,  $\langle \{(\neg a \vee b) \wedge c, \neg b \wedge d\}, \neg a \wedge c \rangle$ ,  $\langle \{\neg a\}, \neg a \rangle$ ,  $\langle \{\neg b \wedge d\}, d \rangle$ .

We now turn to how the notion of connection graphs can be harnessed for focusing the search for arguments. In this section we restrict consideration to clause knowledgebases as follows.

**Definition 2.** A language of clauses  $\mathcal{C}$  is composed from a set of atoms  $\mathcal{A}$  as follows: If  $\alpha$  is an atom, then  $\alpha$  is a **positive literal**, and  $\neg\alpha$  is a **negative literal**. If  $\beta$  is a positive literal, or  $\beta$  is a negative literal, then  $\beta$  is a **literal**. If  $\beta_1, \dots, \beta_n$  are literals, then  $\beta_1 \vee \dots \vee \beta_n$  is a **clause**. A **clause knowledgebase** is a set of clauses.

We introduce relations on the elements of  $\mathcal{C}$ , that will be used to determine the links of graphs. We start by introducing the Disjuncts function which will be used for defining the attack relations between pairs of clauses.

**Definition 3.** The Disjuncts function takes a clause and returns the set of disjuncts in the clause, and hence  $\text{Disjuncts}(\beta_1 \vee \dots \vee \beta_n) = \{\beta_1, \dots, \beta_n\}$ .

**Definition 4.** Let  $\phi$  and  $\psi$  be clauses. Then,  $\text{Preattacks}(\phi, \psi) = \{\beta \mid \beta \in \text{Disjuncts}(\phi)$  and  $\neg\beta \in \text{Disjuncts}(\psi)\}$ .

**Example 2.**  $\text{Preattacks}(a \vee \neg b \vee \neg c \vee d, a \vee b \vee \neg d \vee e) = \{\neg b, d\}$ ,  $\text{Preattacks}(a \vee b \vee \neg d \vee e, a \vee \neg b \vee \neg c \vee d) = \{b, \neg d\}$ ,  $\text{Preattacks}(a \vee b \vee \neg d, a \vee b \vee c) = \emptyset$ ,  $\text{Preattacks}(a \vee b \vee \neg d, a \vee b \vee d) = \{\neg d\}$ ,  $\text{Preattacks}(a \vee b \vee \neg d, e \vee c \vee d) = \{\neg d\}$ .

**Definition 5.** Let  $\phi$  and  $\psi$  be clauses. If  $\text{Preattacks}(\phi, \psi) = \{\beta\}$  for some  $\beta$ , then  $\text{Attacks}(\phi, \psi) = \beta$  otherwise  $\text{Attacks}(\phi, \psi) = \text{null}$ .

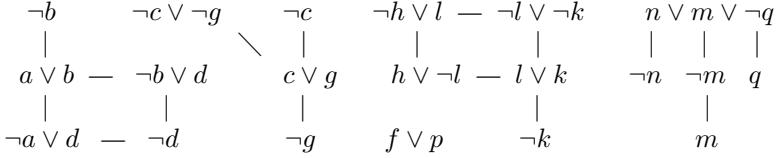
**Example 3.**  $\text{Attacks}(a \vee \neg b \vee \neg c \vee d, a \vee b \vee \neg d \vee e) = \text{null}$ ,  $\text{Attacks}(a \vee b \vee \neg d, a \vee b \vee c) = \text{null}$ ,  $\text{Attacks}(a \vee b \vee \neg d, a \vee b \vee d) = \neg d$ ,  $\text{Attacks}(a \vee b \vee \neg d, e \vee c \vee d) = \neg d$ .

Hence, the Preattacks relation is defined for any pair of clauses  $\phi, \psi$  while the Attacks relation is defined for a pair of clauses  $\phi, \psi$  for which  $|\text{Preattacks}(\phi, \psi)| = 1$ .

We now introduce some types of graphs where each node corresponds to a clause and the links between each pair of clauses are determined according to the binary relations defined above. In the following examples of graphs we use the  $|$ ,  $/$ ,  $\backslash$  and  $-$  symbols to denote arcs in the pictorial representation of a graph.

**Definition 6.** Let  $\Delta$  be a clause knowledgebase. The **connection graph** for  $\Delta$ , denoted  $\text{Connect}(\Delta)$ , is a graph  $(N, A)$  where  $N = \Delta$  and  $A = \{(\phi, \psi) \mid \text{there is a } \beta \in \text{Disjuncts}(\phi) \text{ such that } \beta \in \text{Preattacks}(\phi, \psi)\}$ .

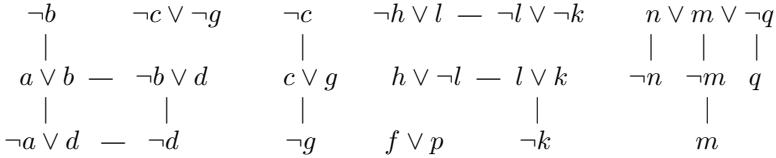
**Example 4.** The following is the connection graph for  $\Delta = \{\neg b, \neg c \vee \neg g, \neg c, f \vee p, \neg l \vee \neg k, a \vee b, \neg b \vee d, c \vee g, \neg h \vee l, l \vee k, \neg a \vee d, \neg d, \neg g, h \vee \neg l, \neg k, n \vee m \vee \neg q, \neg m, \neg n, m, q\}$



The attack graph defined below is a subgraph of the connection graph identified using the Attacks function.

**Definition 7.** Let  $\Delta$  be a clause knowledgebase. The **attack graph** for  $\Delta$ , denoted  $\text{AttackGraph}(\Delta)$ , is a graph  $(N, A)$  where  $N = \Delta$  and  $A = \{(\phi, \psi) \mid \text{there is a } \beta \in \text{Disjuncts}(\phi) \text{ such that } \text{Attacks}(\phi, \psi) = \beta\}$ .

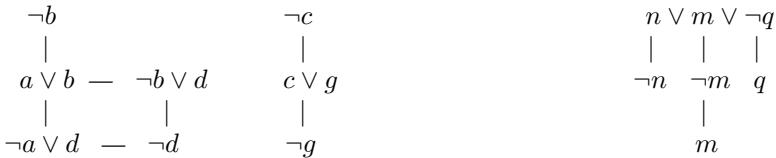
**Example 5.** Continuing Example 4, the following is the attack graph for  $\Delta$ .



The following definition of closed graph introduces a kind of connected subgraph of the attack graph where for each clause  $\phi$  in the subgraph and for each disjunct  $\beta$  in  $\phi$  there is another clause  $\psi$  in the subgraph such that  $\text{Attacks}(\psi, \phi) = \beta$  holds.

**Definition 8.** Let  $\Delta$  be a clause knowledgebase. The **closed graph** for  $\Delta$ , denoted  $\text{Closed}(\Delta)$ , is the largest subgraph  $(N, A)$  of  $\text{AttackGraph}(\Delta)$ , such that for each  $\phi \in N$ , for each  $\beta \in \text{Disjuncts}(\phi)$  there is a  $\psi \in N$  with  $\text{Attacks}(\psi, \phi) = \beta$ .

**Example 6.** Continuing Example 5, the following is the closed graph for  $\Delta$ .

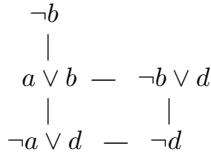


The focal graph (defined next) is a subgraph of the closed graph for  $\Delta$  which is specified by a clause  $\phi$  from  $\Delta$  and corresponds to the part of the closed graph that contains  $\phi$ . In the following, we assume a component of a graph means that each node in the component is connected to any other node in the component by a path.

**Definition 9.** Let  $\Delta$  be a clause knowledgebase and  $\phi$  be a clause in  $\Delta$  which we call the **epicentre**. The **focal graph** of  $\phi$  in  $\Delta$  denoted  $\text{Focal}(\Delta, \phi)$  is defined as follows: If

there is a component  $X$  in  $\text{Closed}(\Delta)$  containing the node  $\phi$ , then  $\text{Focal}(\Delta, \phi) = X$ , otherwise  $\text{Focal}(\Delta, \phi)$  is the empty graph.

**Example 7.** Continuing Example 6, the following is the focal graph of  $\neg b$  in  $\Delta$ ,



The last example illustrates how the notion of the focal graph of an epicentre  $\phi$  in  $\Delta$  can be used in order to focus on the part of the knowledgebase that is relevant to  $\phi$ . Later we will describe why the focal graph is important when it relates to a claim and how it can be used to reduce the search space when looking for arguments from propositional knowledge in conjunctive normal form.

**Proposition 1.** Let  $\Delta$  be a set of clauses. Then,  $\forall \gamma_i, \gamma_j \in \Delta$ , either  $\text{Focal}(\Delta, \gamma_i)$  and  $\text{Focal}(\Delta, \gamma_j)$  are the same component or they are disjoint components.

### 3. Algorithm for finding the focal graph

In this section we present the algorithm (Algorithm 1) that returns the set of nodes of the focal graph of an epicentre  $\phi$  in a clause knowledgebase  $\Delta$ . The  $\text{GetFocal}(\Delta, \phi)$  algorithm finds the focal graph of  $\phi$  in  $\Delta$  by a depth first search following the links of the component of the  $\text{AttackGraph}(\Delta)$  that is linked to  $\phi$ . For this, we have a data structure  $\text{Node}_\psi$  (for each  $\psi \in \Delta$ ) that represents the node for  $\psi$  in  $\text{AttackGraph}(\Delta)$ . The attack graph can be represented by an adjacency matrix. Initially all the nodes are allowed as candidate nodes for the focal graph of  $\phi$  in  $\Delta$ , and then during the search they can be rejected if they do not satisfy the conditions of the definition for the focal graph. The algorithm chooses the appropriate nodes by using the boolean method  $\text{isConnected}(C, \text{Node}_\psi)$  which tests whether a node  $\text{Node}_\psi$  of the attack graph  $C = (N, A)$  is such that each literal  $\beta \in \text{Disjuncts}(\psi)$  corresponds to at least one arc to an allowed node (i.e.  $\forall \beta \in \text{Disjuncts}(\psi), \exists \text{Node}_{\psi'} \in \text{AllowedNodes}$  s.t.  $\text{Attacks}(\psi, \psi') = \beta$ ), and so it returns false when there is a  $\beta \in \text{Disjuncts}(\psi)$  for which there is no  $\text{Node}_{\psi'} \in \text{AllowedNodes}$  s.t.  $\text{Attacks}(\psi, \psi') = \beta$ . If this method does return false for a  $\text{Node}_\psi$ , then  $\text{Node}_\psi$  is rejected and the algorithm backtracks to retest whether its adjacent allowed nodes are still connected. If some of them are no longer connected, they are rejected and in the same way their adjacent allowed nodes are tested recursively.

In the next section we show how this algorithm can be used for the generalised problem of finding arguments for any propositional formula when expressed in conjunctive normal form.

### 4. Using the focal graph algorithm for formulae in conjunctive normal form

To explain how the restricted language of clauses and Algorithm 1, can be used to deal with formulae (and thereby extend the proposal in [10]) we will first give some new

**Algorithm 1** GetFocal( $\Delta, \phi$ )

---

```

Let  $C = (N, A)$  be the attack graph for  $\Delta$  and  $\phi$ 
Let AllowedNodes =  $\{Node_\psi \mid \psi \in N\}$ 
Let VisitedNodes be the empty set.

if  $\phi \notin \Delta$  or  $\neg$ isConnected( $C, Node_\phi$ ) then
    return  $\emptyset$ 
else
    Let S be an empty Stack
    push  $Node_\phi$  onto S
end if

while S is not empty do
    Let  $Node_\psi$  be the top of the stack S
    if  $Node_\psi \in$  AllowedNodes then
        if isConnected( $C, Node_\psi$ ) then
            if  $Node_\psi \in$  VisitedNodes then
                pop  $Node_\psi$  from S
            else
                VisitedNodes = VisitedNodes  $\cup \{Node_\psi\}$ 
                pop  $Node_\psi$  from S
                for all  $Node_{\psi'} \in$  AllowedNodes with Attacks( $\psi, \psi'$ )  $\neq null$  do
                    push  $Node_{\psi'}$  onto S
                end for
            end if
        end if
    else
        AllowedNodes = AllowedNodes  $\setminus \{Node_\psi\}$ 
        VisitedNodes = VisitedNodes  $\cup \{Node_\psi\}$ 
        pop  $Node_\psi$  from S.
        for all  $Node_{\psi'} \in (\text{AllowedNodes} \setminus \text{VisitedNodes})$  with Attacks( $\psi, \psi'$ )  $\neq null$  do
            push  $Node_{\psi'}$  onto S
        end for
    end if
end while
return AllowedNodes  $\cap$  VisitedNodes

```

---

subsidiary definitions. For the following we assume a formula is in conjunctive normal form and a set of formulae contains formulae in conjunctive normal form.

**Definition 10.** Let  $\psi = \gamma_1 \wedge \dots \wedge \gamma_n$  be a formula. The Conjuncts( $\psi$ ) function returns the clause knowledgebase  $\{\gamma_1, \dots, \gamma_n\}$ .

**Example 8.** For  $\phi = (a \vee b) \wedge (a \vee d \vee \neg c) \wedge \neg e$ , Conjuncts( $\phi$ ) =  $\{a \vee b, a \vee d \vee \neg c, \neg e\}$ .

**Definition 11.** Let  $\Phi = \{\phi_1, \dots, \phi_k\}$  be a set of formulae. The SetConjuncts( $\Phi$ ) function returns the union of all the conjuncts of the formulae from the set: SetConjuncts( $\Phi$ ) =

$$\bigcup_{\phi_i \in \Phi} \text{Conjuncts}(\phi_i).$$

**Example 9.** For  $\Phi = \{\neg a, (a \vee b) \wedge \neg d, (c \vee d) \wedge (e \vee f \vee \neg g), \neg d\}$ ,  $\text{SetConjuncts}(\Phi) = \{\neg a, a \vee b, \neg d, c \vee d, e \vee f \vee \neg g\}$ .

Let  $\psi = \delta_1 \wedge \dots \wedge \delta_n$  be a formula and let  $\bar{\psi}$  denote the conjunctive normal form of the negation of  $\psi$ , and so  $\bar{\psi} = \gamma_1 \wedge \dots \wedge \gamma_m \equiv \neg \psi$ . Then, if we seek supports for arguments for  $\psi$  from a knowledgebase  $\Phi = \{\phi_1, \dots, \phi_k\}$ , instead of searching among the arbitrary subsets of  $\Phi$ , we can search among the subsets of  $\Phi$  that consist of formulae whose conjuncts are contained in one of the focal graphs of each  $\gamma_i$  in  $\text{SetConjuncts}(\Phi \cup \{\bar{\psi}\})$ . For this we need the notion of the SubFocus and the SupportBase defined next.

**Definition 12.** Let  $\Phi$  be a knowledgebase and  $\psi \in \Phi$ . Then for each  $\gamma_i \in \text{Conjuncts}(\psi)$ ,  $\text{SubFocus}(\Phi, \gamma_i) = \text{Focal}(\text{SetConjuncts}(\Phi), \gamma_i)$ .

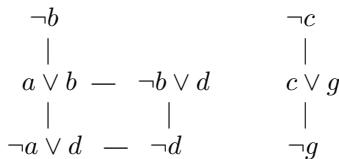
**Example 10.** Let  $\Phi = \{(a \vee b) \wedge (f \vee p) \wedge \neg c, (\neg a \vee d) \wedge (\neg c \vee \neg g), \neg d, \neg d \wedge (\neg h \vee l), q \wedge (\neg h \vee l), c \vee g, \neg g, \neg b, \neg b \vee d, l \vee k, m \wedge (\neg l \vee \neg k), \neg k \wedge (n \vee m \vee \neg q), h \vee \neg l, \neg m \wedge \neg n, m \wedge q\}$ . Then,  $\text{Conjuncts}(\Phi)$  is equal to  $\Delta$  from example 4. Let  $\phi = (a \vee b) \wedge (f \vee p) \wedge \neg c$ , and let  $\gamma_1$  denote  $a \vee b$ ,  $\gamma_2$  denote  $f \vee p$  and  $\gamma_3$  denote  $\neg c$ . So, if  $\text{SubFocus}(\Phi, \gamma_1) = (N_1, A_1)$ ,  $\text{SubFocus}(\Phi, \gamma_2) = (N_2, A_2)$  and  $\text{SubFocus}(\Phi, \gamma_3) = (N_3, A_3)$  then  $N_1 = \{a \vee b, \neg a \vee d, \neg d, \neg b, \neg b \vee d\}$ ,  $N_2 = \emptyset$ , and  $N_3 = \{\neg c, c \vee g, \neg g\}$ .

The following definition introduces the notion of the query graph of a formula  $\psi$  in a knowledgebase  $\Phi$ , which is a graph consisting of all the subfocuses of each of the  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$  in  $\text{Conjuncts}(\Phi \cup \{\bar{\psi}\})$ . For a graph  $C = (N, A)$  we let the function  $\text{Nodes}(C)$  return the set of clauses corresponding to the nodes of the graph (i.e.  $\text{Nodes}(C) = N$ ).

**Definition 13.** Let  $\Phi$  be a knowledgebase and  $\psi$  be a formula. The **query graph** of  $\psi$  in  $\Phi$  denoted  $\text{Query}(\Phi, \psi)$  is the closed graph for the nodes

$$\bigcup_{\gamma_i \in \text{Conjuncts}(\bar{\psi})} \text{Nodes}(\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_i))$$

**Example 11.** Let  $\Phi' = \{(\neg a \vee d) \wedge (\neg c \vee \neg g), \neg d, \neg d \wedge (\neg h \vee l), q \wedge (\neg h \vee l), c \vee g, \neg g, \neg b, \neg b \vee d, l \vee k, m \wedge (\neg l \vee \neg k), \neg k \wedge (n \vee m \vee \neg q), (h \vee \neg l), \neg m \wedge \neg n, m \wedge q\}$  and let  $\psi = (\neg a \vee \neg f \vee c) \wedge (\neg a \vee \neg p \vee c) \wedge (\neg b \vee \neg f \vee c) \wedge (\neg b \vee \neg p \vee c)$ . For  $\Phi'$  and  $\psi$  we have  $\bar{\psi} = (a \vee b) \wedge (f \vee p) \wedge \neg c$ , which is equal to  $\phi$  from Example 10 and  $\Phi' \cup \{\bar{\psi}\}$  is equal to  $\Phi$  from Example 10. Hence, continuing Example 10, the query graph of  $\psi$  in  $\Phi$  is presented below and consists of the subgraphs  $(N_1, A_1)$ ,  $(N_2, A_2)$  and  $(N_3, A_3)$ .



Hence, using each of the conjuncts  $\gamma_i$  of  $\bar{\psi}$  as the epicentre for the focal graph in  $\text{SetConjuncts}(\Phi \cup \{\bar{\psi}\})$  we obtain the components of the query graph of  $\psi$  in  $\Phi$ . The following definition introduces the notion of a zone, which relates each clause from each such subfocus to one or more formulae from knowledgebase  $\Phi$ .

**Definition 14.** Let  $\Phi$  be a knowledgebase and  $\psi$  be a formula. Then, for each  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ ,

$$\text{Zone}(\Phi, \gamma_i) = \{\phi \in \Phi \mid \text{Conjuncts}(\phi) \cap \text{Nodes}(\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_i)) \neq \emptyset\}.$$

**Example 12.** Continuing Example 11, for each  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ ,  $i = 1 \dots 3$  we have  $\text{Zone}(\Phi', \gamma_1) = \{(\neg a \vee d) \wedge (\neg c \vee \neg g), \neg d, \neg b, \neg b \vee d, \neg d \wedge (\neg h \vee l)\}$ ,  $\text{Zone}(\Phi', \gamma_2) = \emptyset$  and  $\text{Zone}(\Phi', \gamma_3) = \{c \vee g, \neg g\}$ .

The supportbase defined next, relates the clauses from the query graph of a formula  $\psi$  in a knowledgebase  $\Phi$  to the corresponding set of formulae from  $\Phi$ .

**Definition 15.** For a knowledgebase  $\Phi$  and a formula  $\psi$  the supportbase is given as follows:

$$\text{SupportBase}(\Phi, \psi) = \bigcup_{\gamma_i \in \text{Conjuncts}(\bar{\psi})} \text{Zone}(\Phi, \gamma_i)$$

**Example 13.** Continuing Example 11,  $\text{SupportBase}(\Phi, \psi) = \{(\neg a \vee d) \wedge (\neg c \vee \neg g), \neg d, \neg b, \neg b \vee d, \neg d \wedge (\neg h \vee l), c \vee g, \neg g\}$

According to the following proposition, the  $\text{SupportBase}(\Phi, \psi)$  defined above is the knowledgebase that contains all the arguments for  $\psi$  from  $\Phi$ .

**Proposition 2.** Let  $\Phi$  be a knowledgebase and  $\psi$  be a formula. If  $\langle \Psi, \psi \rangle$  is an argument from  $\Phi$  and  $\Psi \subseteq \Phi$ , then  $\Psi \subseteq \text{SupportBase}(\Phi, \psi)$ .

Hence, by Proposition 2 it follows that instead of using the power set of the initial knowledgebase  $\Phi$  in order to look for arguments for  $\psi$ , we can use the power set of  $\text{SupportBase}(\Phi, \psi)$ . Using these definitions, we can introduce the additional algorithms that delineate the extension of the language of clauses used in previous sections and the use of Algorithm 1 to support a language of propositional formulae in conjunctive normal form. These are Algorithm 2 and Algorithm 3. Since our problem is focused on searching for arguments for a claim  $\psi$ , the part of the knowledgebase that we want to isolate will be delineated by  $\psi$  and, in particular, by the conjuncts of  $\bar{\psi}$ . Algorithm 2 returns the query graph of  $\psi$  in  $\Phi$  as a set containing all the  $\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_i)$  components for each  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ . Then, Algorithm 3 uses these results in order to retrieve the set of formulae from the initial knowledgebase to which each such set of conjuncts corresponds. So, Algorithm 3 returns a set of sets, each of which represents a zone for each  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ . The union of all these sets will be  $\text{SupportBase}(\Phi, \psi)$ .

In Algorithm 2 we can see that it is not always necessary to use Algorithm 1 for each of the  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$  when trying to isolate the appropriate subsets of  $\Phi$ . Testing for containment of a clause  $\gamma_j \in \text{Conjuncts}(\bar{\psi})$  in an already retrieved set  $\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_i)$ ,  $i < j$  (where the ordering of the indices describes the order in which the algorithm is applied for each of the conjuncts) is enough to give the  $\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_j)$  according to proposition 1. Furthermore, according to the following proposition, conjuncts of  $\bar{\psi}$  within the same  $\text{SubFocus}$  correspond to the same set of formulae from  $\Phi$ .

**Algorithm 2** GetQueryGraph( $\Phi, \psi$ )

---

Let  $\bar{\psi}$  be  $\neg\psi$  in CNF :  $\bar{\psi} \equiv \gamma_1 \wedge \dots \wedge \gamma_m$   
 Let  $S$  be a set to store sets of clauses, initially empty  
 Let  $\text{Clauses} = \text{SetConjuncts}(\Phi \cup \{\bar{\psi}\})$   
**for**  $i = 1 \dots m$  **do**  
   **if**  $\exists S_j \in S$  s.t.  $\gamma_i \in S_j$  **then**  
      $i = i + 1$   
   **else**  
      $S_i = \text{GetFocal}(\text{Clauses}, \gamma_i)$   
   **end if**  
    $S = S \cup \{S_i\}$   
**end for**  
**return**  $S$

---

**Proposition 3.** *Let  $\Phi$  be a set of formulae and let  $\psi$  be a formula. If  $\text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_i) = \text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \gamma_j)$  for some  $\gamma_i, \gamma_j \in \text{Conjuncts}(\bar{\psi})$ , then  $\text{Zone}(\Phi, \gamma_i) = \text{Zone}(\Phi, \gamma_j)$ .*

The converse of the last proposition does not hold as the following example illustrates.

**Example 14.** *For  $\Phi = \{(c \vee g) \wedge d, d \vee f, \neg q, (d \vee p) \wedge f, \neg n, k \vee \neg m\}$  and  $\psi = c \vee g \vee d$  we have  $\bar{\psi} = \neg c \wedge \neg g \wedge \neg d$  and  $N_1 \equiv \text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \neg c) = \{\neg c, \neg g, c \vee g\}$ ,  $N_2 \equiv \text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \neg g) = \{\neg c, \neg g, c \vee g\} = N_1$  and  $N_3 \equiv \text{SubFocus}(\Phi \cup \{\bar{\psi}\}, \neg d) = \{\neg d, d\}$ . Furthermore,  $\text{Zone}(\Phi, \neg c) = \text{Zone}(\Phi, \neg g) = \text{Zone}(\Phi, \neg d) = \{(c \vee g) \wedge d\}$  although  $N_3 \neq N_2$  and  $N_3 \neq N_1$ .*

Hence, conjuncts of  $\bar{\psi}$  with the same focal graph correspond to the same support-base. Taking this into account, the following algorithm retrieves all the supportbases for each  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ .

**Algorithm 3** RetrieveZones( $\Phi, \psi$ )

---

Let  $Z$  be a set to store sets of formulae, initially empty  
 Let  $S = \text{GetQueryGraph}(\Phi, \psi) \equiv \{S_1, \dots, S_k\}$   
**for**  $i = 1 \dots k$  **do**  
   Let  $Z_i$  be the emptyset.  
   **for**  $j = 1 \dots |S_i|$  **do**  
     Let  $\gamma_j$  be the j-th element of  $S_i$   
     Let  $C_j = \{\phi \in \Phi \mid \gamma_j \in \text{Conjuncts}(\phi)\}$   
      $Z_i = Z_i \cup C_j$   
   **end for**  
    $Z = Z \cup \{Z_i\}$   
**end for**  
**return**  $Z$

---

So, Algorithm 3 returns a set of sets, corresponding to all the possible zones identified by all the  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$ . The union of all these sets will be  $\text{SupportBase}(\Phi, \psi)$ .

Instead of looking for arguments for  $\psi$  among the arbitrary subsets of  $\text{SupportBase}(\Phi, \psi)$ , we can search the power set of each non empty  $\text{Zone}(\Phi, \gamma_i)$  separately as the following proposition indicates.

**Proposition 4.** *Let  $\Phi$  be a knowledgebase and  $\psi$  be a clause. If  $\langle \Psi, \psi \rangle$  is an argument from  $\Phi$ , then there is a  $\gamma_i \in \text{Conjuncts}(\bar{\psi})$  s.t  $\Psi \subseteq \text{Zone}(\Phi, \gamma_i)$ .*

In future work we plan to address the efficiency of using the subsets of each zone separately instead of using the power set of the supportbase in our search for arguments. We conjecture that this will improve the performance on average for finding arguments.

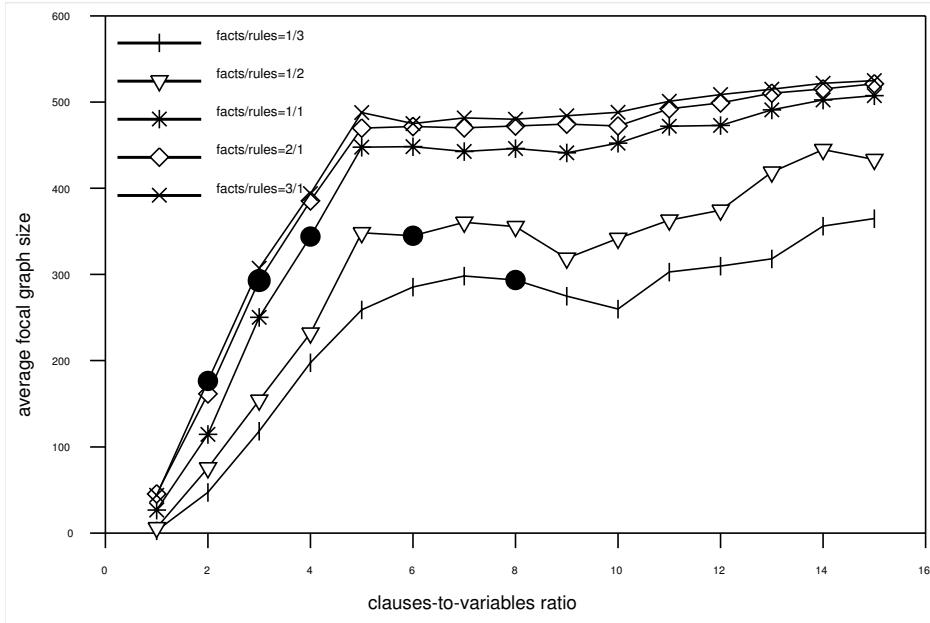
## 5. Experimental results

This section covers a preliminary experimental evaluation of algorithm 1 using a prototype implementation programmed in java running on a modest PC (Core2 Duo 1.8GHz).

The experimental data were obtained using randomly generated clause knowledgebases of a fixed number of 600 clauses according to the fixed clause length model K-SAT ([19,13]) where the chosen length (i.e. K) for each clause was either a disjunction of 3 literals or a disjunction of 1 literal. The clauses of length 3 can be regarded as **rules** and clauses of length 1 as **facts**. Each disjunct of each clause was randomly chosen out of a set of  $N$  distinct variables (i.e. atoms) and negated with probability 0.5.

In the experiment, we considered two dimensions. The first dimension was the clauses-to-variables ratio. For the definition of this ratio we take the integer part of the division of the number of clauses in  $\Delta$  by the number of variables  $N$  (i.e.  $\lfloor |\Delta|/|N| \rfloor$ ). The second dimension was the proportion of facts-to-rules in the knowledgebases tested. The preliminary results are presented in Figure 1 where each curve in the graph corresponds to one of those variations on the proportion of facts-to-rules. More precisely, each curve relates to one of the following  $(n, n')$  tuples where  $n$  represents the number of facts and  $n'$  represents the number of rules in the set: (150,450), (200,400), (300,300), (400,200), (450,150). Since each clause knowledgebase contains 600 elements, each of these tuples sums to 600. Each point on each curve is the average focal graph size from 1000 repetitions of running Algorithm 1 for randomly generated epicentres and randomly generated knowledgebases of a fixed clauses-to-variables ratio represented by coordinate  $x$ . For the results presented, since the values on axis  $x$  ranges from 1 to 15, the smallest number of variables used throughout the experiment was 40 which corresponds to a clauses-to-variables ratio of 15, while the largest number of variables used was 600 which corresponds to a clauses-to-variables ratio of 1.

The evaluation of our experiment was based on the size of the focal graph of an epicentre in a clause knowledgebase compared to the cardinality of the knowledgebase. For a fixed number of clauses, the number of distinct variables that occur in the disjuncts of all these clauses determines the size of the focal graph. In Figure 1 we see that as the clauses-to-variables ratio increases, the average focal graph size also increases because an increasing clauses-to-variables ratio for a fixed number of clauses implies a decreasing number of variables and this allows for a distribution of the variables amongst the clauses such that it is more likely for a literal to occur in a clause with its opposite occurring in another clause. We have noticed in previous experiments [10] that for a clause knowledgebase consisting of 3-place clauses only, an increasing clauses-to-variables ra-



**Figure 1.** Focal graph size variation with the clauses-to-variables ratio

tio in the range [5, 10] ameliorates the performance of the system as it increases the probability of a pair of clauses  $\phi, \psi$  from  $\Delta$  being such that  $|\text{Preattacks}(\phi, \psi)| > 1$ . This is because a ratio in this range makes the occurrence of a variable and its negation in the clauses of the set so frequent that it allows the Attacks relation to be defined only on a small set of clauses from the randomly generated clause knowledgebase. In the graph presented in this paper though, this is less likely to happen as the clause knowledgebases tested involve both facts and rules.

Including literals (facts) in the knowledgebases used during the experiment makes the repeated occurrence of the same fact and its complement in the randomly generated clause knowledgebases, and hence in the subsequent focal graph, more frequent. It is for this reason that the curves with lower facts-to-rules proportion have a lower average focal graph (for each clauses-to-variables ratio). The symbol  $\bullet$  on each of the curves indicates the highest possible clauses-to-variables ratio that would allow for a randomly generated clause knowledgebase consisting of the corresponding proportion of facts and rules to contain only distinct elements. Hence, for the data presented in this graph, the largest average focal graph of a randomly generated clause in a randomly generated clause knowledgebase of 600 distinct elements has the value 343.99 which corresponds to 57% of the initial knowledgebase. The values of the parameters with which this maximum is obtained correspond to a clauses-to-variables ratio equal to 4 on knowledgebases with a 1 to 1 proportion of facts-to-rules.

The average time for each repetition of the algorithm ranged from 6.3 seconds (for a facts-to-rules proportion of 1-3) to 13.8 seconds (for a facts-to-rules proportion of 3-1). So, the results show that for an inexpensive process we can substantially reduce the search space for arguments.

## 6. Discussion

In this paper, we have proposed the use of a connection graph approach as a way of ameliorating the computation cost when searching for arguments. We have extended the theory and algorithms proposed in previous work [10] for a language of clauses so as to deal with any set of propositional formulae provided that these are represented in conjunctive normal form. We have provided theoretical results to ensure the correctness of the proposal, and we have provided provisional empirical results to indicate the potential advantages of the approach.

## References

- [1] L. Amgoud and C. Cayrol. A model of reasoning based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–216, 2002.
- [2] S. Benferhat, D. Dubois, and H. Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In *Proceedings of the 9th Annual Conference on Uncertainty in Artificial Intelligence (UAI 1993)*, pages 1449–1445. Morgan Kaufmann, 1993.
- [3] Ph. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128:203–235, 2001.
- [4] Ph. Besnard and A. Hunter. *Elements of Argumentation*. MIT Press, 2008.
- [5] D. Bryant, P. Krause, and G. Vreeswijk. Argue tuProlog: A lightweight argumentation engine for agent applications. In *Computational Models of Argument (Comma'06)*, pages 27–32. IOS Press, 2006.
- [6] C. Cayrol, S. Doutre, and J. Mengin. Dialectical proof theories for the credulous preferred semantics of argumentation frameworks. In *Quantitative and Qualitative Approaches to Reasoning with Uncertainty*, volume 2143 of *LNCS*, pages 668–679. Springer, 2001.
- [7] C. Chesñevar, A. Maguitman, and R. Loui. Logical models of argument. *ACM Computing Surveys*, 32:337–383, 2000.
- [8] Y. Dimopoulos, B. Nebel, and F. Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141:57–78, 2002.
- [9] P. Dung, R. Kowalski, and F. Toni. Dialectical proof procedures for assumption-based admissible argumentation. *Artificial Intelligence*, 170:114–159, 2006.
- [10] V. Efstathiou and A. Hunter. Algorithms for effective argumentation in classical propositional logic. In *Proceedings of the International Symposium on Foundations of Information and Knowledge Systems (FOIKS 2008)*, LNCS. Springer, 2008.
- [11] M. Elvang-Göransson, P. Krause, and J. Fox. Dialectic reasoning with classically inconsistent information. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI 1993)*, pages 114–121. Morgan Kaufmann, 1993.
- [12] A. García and G. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.
- [13] I. P. Gent and T. Walsh. Easy problems are sometimes hard. *Artificial Intelligence*, 70(1-2):335–345, 1994.
- [14] A. Kakas and F. Toni. Computing argumentation in logic programming. *Journal of Logic and Computation*, 9:515–562, 1999.
- [15] R. Kowalski. A proof procedure using connection graphs. *Journal of the ACM*, 22:572–595, 1975.
- [16] R. Kowalski. *Logic for problem solving*. North-Holland Publishing, 1979.
- [17] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7:25–75, 1997.
- [18] H. Prakken and G. Vreeswijk. Logical systems for defeasible argumentation. In D. Gabbay, editor, *Handbook of Philosophical Logic*. Kluwer, 2000.
- [19] B. Selman, D. G. Mitchell, and H. J. Levesque. Generating hard satisfiability problems. *Artificial Intelligence*, 81(1-2):17–29, 1996.
- [20] G. Vreeswijk. An algorithm to compute minimally grounded and admissible defence sets in argument systems. In *Computational Models of Argument (Comma'06)*, pages 109–120. IOS Press, 2006.

# Decision Rules and Arguments in Defeasible Decision Making

Edgardo FERRETTI<sup>a</sup>, Marcelo L. ERRECALDE<sup>a</sup>,  
Alejandro J. GARCÍA<sup>b,c</sup> and Guillermo R. SIMARI<sup>c,1</sup>

<sup>a</sup> Department of Computer Science, Universidad Nacional de San Luis,  
San Luis - Argentina

<sup>b</sup> Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

<sup>c</sup> Department of Computer Science and Engineering,  
Universidad Nacional del Sur, Bahía Blanca - Argentina

**Abstract.** In this paper we present a model for defeasible decision making that combines decision rules and arguments. In this decision framework we can change the agent's decision policy in a flexible way, with minor changes in the criteria that influence the agent's preferences and the comparison of arguments. Our approach includes a simple methodology for developing the decision components of the agent. A decision framework designed with this methodology exhibits some interesting properties. If the agent (decision maker) has available all the relevant knowledge about its preferences among the different alternatives that could be conceivably posed to it, then our proposal implements a *rational preference relation*. In opposition, if the agent has partial knowledge about its preferences, the decisions made by the agent still exhibits a behavior consistent with the *weak axiom of revealed preference* of the *choice-based approach*, a more flexible approach to Individual Decision Making than the *preference-based approach*. The principles stated in this work are exemplified in a robotic domain, where a robot should make decisions about which box must be transported next.

**Keywords.** Individual Decision Making, Defeasible Decision Making, DeLP.

## Introduction

Classical approaches to decision making are usually based in the principle that the objectives of the decision maker are summarized in a *rational preference relation* or a *utility function* that represents this relation. These approaches have a solid theoretical foundation but some weaknesses can be observed when applied to diverse real world problems. One of the main misstatements is that it is not always straight to understand the general process by which an agent (decision maker) concludes that an alternative is better than another. Besides, neither is clear which should be the agent's behavior when the information referred to its preferences is incomplete or contradictory, a very common situation in real world problems.

---

<sup>1</sup>Corresponding Author: Guillermo R. Simari, Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Av. Alem 1253, (B8000CPB) Bahía Blanca, Argentina; E-mail: grs@cs.uns.edu.ar

In this work we present a first approach based on defeasible decision making to deal with these limitations. At this end, in our proposal, the reasons by which an alternative will be deemed better than another, are explicitly considered in the argumentation process involved in warranting information from a *Defeasible Logic Program* [1]. We propose a modular approach that allows to define in a direct way the criteria used to compare arguments, and even to easily modify these criteria. The information warranted by the dialectical process is then used in decision rules that implement the agent's general decision making policy. Basically, decision rules establish patterns of behavior of the agent specifying under which conditions a set of alternatives will be considered acceptable.

Our proposal includes a simple methodology for programming the components involved in the agent's decision process. Besides it is demonstrated that programming the knowledge base of an agent following this methodology will exhibit some interesting guarantees with respect to the selected alternatives. When all the relevant information about the agent's preferences is specified, its choice behavior will be equivalent to the decisions based on a *rational preference relation* [2]. On the other hand, if some aspects related to the agent's preferences are not completely specified, the agent will still be capable of exhibiting a coherent decision behavior. In this case, we make explicit the connections between our proposal and the *choice rules* approach to decision making [2], an approach that leaves room, in principle, for more general forms of individual behavior than the one possible with the preference-based approach.

The principles stated in this work are exemplified in a robotic domain where a robot should make decisions about which box must be transported next. It is a simple scenario where a *Khepera 2* robot [3] collects boxes scattered through the environment.<sup>2</sup> The environment (see Figure 1(a)) consists of a square arena of 100 units per side which is conceptually divided into square cells of 10 units per side each. A global camera provides to the robot the necessary information to perform its activities. The *store* is a  $30 \times 30$  units square on the top-right corner and represents the target area where the boxes should be transported. There are boxes of three different sizes (*small*, *medium* and *big*) spread over the environment.

The autonomy of the robot is limited and it cannot measure the state of its battery, thus, the robot cannot perform a globally optimized task. Because of this drawback, a *greedy* strategy is used to select the next box. Therefore, the robot will *prefer its nearer boxes, then the boxes nearer to the store, and finally the smaller ones*. These preferences will be explicitly established using a preference order among the literals that represent them. To perform the reasoning, the robot will use the perceptual information about the world, and its preferences which will be represented with strict and defeasible rules. Arguments for and against selecting a box will be considered in order to select the more appropriate one.

The paper is organized as follows. Section 1 introduces the theory of individual decision making where two related approaches to model the agent's decision are considered. In Section 2 we present our approach to *Defeasible Decision Making*: a formalism used for knowledge representation and reasoning is provided and, decision rules and a literal-based comparison criterion are defined. Then, Section 3 relates our implementation of decision making with the approaches presented in Section 1. Finally, Section 4 offers some conclusions and briefly describes some related work.

---

<sup>2</sup>To perform simulations we use *Webots* [4], a 3D realistic professional simulator. In [5,6] a framework that allows the Khepera robots to perform argumentative reasoning was presented.

## 1. Individual Decision Making

This section introduces the theory of individual decision making presenting an excerpt of Chapter 1 of [2], where two related approaches to model the agent's decision are considered. These approaches are described below in two separate subsections.

### 1.1. Preference Relations

The starting point for any individual decision problem is a set of possible (mutually exclusive) *alternatives* from which the decision maker (an agent in our case) must choose. In the discussion that follows, we denote this set of alternatives by  $X$ .

In classical decision making domains, it is usually assumed that the agent's choice behavior is modeled with a binary *preference relation*  $\succsim$ , where given  $\{x, y\} \subseteq X$ ,  $x \succsim y$  means that “ $x$  is at least as good as  $y$ ”. From  $\succsim$  we can derive two other important relations:

- The *strict preference* relation  $\succ$ , defined as  $x \succ y \Leftrightarrow x \succsim y$  but not  $y \succsim x$  and read “ $x$  is preferred to  $y$ ”;
- The *indifference* relation  $\sim$ , defined as  $x \sim y \Leftrightarrow x \succsim y$  and  $y \succsim x$  and read “ $x$  is indifferent to  $y$ ”.

It is common to require the preference relation  $\succsim$  to be *rational* (see Definition 1) and this is a necessary condition if  $\succsim$  will be represented by a utility function. The hypothesis of rationality is embodied in two basic assumptions about the preference relation  $\succsim$ , as defined next.

**Definition 1** A preference relation  $\succsim$  is rational if it possesses the following two properties:

1. *Completeness*: for all  $x, y \in X$ , we have that  $x \succsim y$  or  $y \succsim x$  (or both).
2. *Transitivity*: for all  $x, y, z \in X$ , if  $x \succsim y$  and  $y \succsim z$ , then  $x \succsim z$ .

The assumption that  $\succsim$  is complete states that the agent has a well-defined preference between any two possible alternatives. Besides, transitivity implies that it is impossible to face the decision maker with a sequence of pairwise choices in which his preferences appear to cycle. Despite of the relevance of this *preference-based approach* (PBA) from a theoretical point of view, it is well known in the decision theory community that completeness and transitivity assumptions are usually very hard to satisfy in real world agents, when evaluating alternatives that are far from the realm of common experience [2].

In our approach, we are interested in defeasible decision making in dynamic domains where new and changing information has to be considered during the decision making process. For this reason, the *static* properties of the agent's preferences are not so important, and should be useful to directly consider the agent's *choice behavior* and to analyze the *dynamics* of the decision processes, when new alternatives have to be considered due to changes in the environment or when new information is available to the agent. From this point of view, an interesting and more flexible formal model of theory of decision making, called *choice-based approach* [2] provides us more adequate tools for evaluating the dynamics involved in the agent's decisions. This formal model of theory of decision making is described in the following subsection.

## 1.2. Choice Rules

The *choice-based approach* (CBA) takes as primitive object the choice behavior of the individual, which is represented by means of a *choice structure*  $(\mathcal{B}, C(\cdot))$  consisting of two elements:

- $\mathcal{B}$  is a set of subsets of  $X$ . Intuitively, each set  $B \in \mathcal{B}$  represents a set of alternatives (or *choice experiment*) that can be conceivably posed to the decision maker. In this way, if  $X = \{x, y, z\}$  and  $\mathcal{B} = \{\{x, y\}, \{x, y, z\}\}$  we will assume that the sets  $\{x, y\}$  and  $\{x, y, z\}$  are valid choice experiments to be presented to the decision maker.
- $C(\cdot)$  is a *choice rule* (a correspondence,  $C : \mathcal{B} \mapsto B$ ) which basically assigns to each set of alternatives  $B \in \mathcal{B}$  a nonempty set that represents the alternatives that the decision maker *might* choose when presented the alternatives in  $B$ . Note that  $C(B) \subseteq B$  for every  $B \in \mathcal{B}$ . When  $C(B)$  contains a single element, this element represents the *individual's choice* from among the alternatives in  $B$ . The set  $C(B)$  might, however, contain more than one element and in this case they would represent the *acceptable alternatives* in  $B$  for the agent.

Similar to the rationality assumption of PBA (Definition 1), in the CBA there is a central assumption called the *weak axiom of revealed preference* (or WARP for short). As we will explain next, this axiom imposes an element of consistency on choice behavior that is similar to the rationality assumptions of the PBA. This WARP axiom defined in [2] is recalled below:

**Definition 2** A choice structure  $(\mathcal{B}, C(\cdot))$  satisfies the weak axiom of revealed preference (WARP) if the following property holds:

If for some  $B \in \mathcal{B}$  with  $x, y \in B$  we have  $x \in C(B)$ , then for any  $B' \in \mathcal{B}$  with  $x, y \in B'$  and  $y \in C(B')$ , we must also have  $x \in C(B')$ .

The weak axiom postulates that if there is some choice experiment  $B \in \mathcal{B}$  such that  $x$  and  $y$  are presented as alternatives ( $x, y \in B$ ) and “ $x$  is revealed at least as good as  $y$ ” (i.e.,  $x \in C(B)$ ) then it does not exist other choice experiment  $B' \in \mathcal{B}$  where “ $y$  is revealed preferred to  $x$ ” (i.e.,  $x, y \in B'$ ,  $y \in C(B')$  and  $x \notin C(B')$ ).

**Example 1** Consider the scenario depicted in Figure 1(a) where there is a robot (*khep1*) and five boxes. Three of the boxes ( $box_1, box_2, box_3$ ) have the same properties: they are small, they are far from the robot and from the store. Box  $box_4$  is medium size, is near to the store and far from *khep1*; and  $box_5$  is big, is near to the store and near to the robot. Consider a set  $\mathcal{B} = \{B_{1.1}, B_{1.2}\}$  of choice experiments composed by two subsets of  $X$ ,  $B_{1.1} = \{box_3, box_4, box_5\}$  and  $B_{1.2} = \{box_1, box_2, box_3, box_4, box_5\}$ . (Note: In the application examples presented in this work, the set of all the alternatives will be  $X = \{box_1, box_2, box_3, box_4, box_5\}$ . In this case, with the aim of exemplifying all the concepts previously mentioned, we decided to include in  $\mathcal{B}$  only two choice experiments, however, more possibilities could be considered.)

Taking into account the preferences of *khep1* (informally presented in the introductory section of this paper, and later formalized in Figure 3),  $C(B_{1.1}) = \{box_5\}$ , because despite of being a big box, is near to the store and is also the only box near to the robot. In the same way,  $C(B_{1.2}) = \{box_5\}$ , satisfying the WARP principle.

Intuitively, the WARP principle reflects the expectation that an individual's observed choices will display a certain amount of consistency [2]. For instance, in Example 1, when *khep1* faces the alternatives  $B_{1.1} = \{box_3, box_4, box_5\}$ , it chooses alternative  $box_5$  and only that, however, we would be surprised to see it choosing  $box_3$  or  $box_4$  when faced with the set of alternatives  $\{box_1, box_2, box_3, box_4, box_5\}$ .

## 2. Defeasible Decision Making

In this section, we introduce the defeasible argumentation formalism that the agent will use for knowledge representation and reasoning. Then, we present the literal-based criterion used to decide between conflicting arguments and finally, the decision rules used to implement the agent's decision making policy are defined (Definition 5). Based on these decision rules, the agent's set of acceptable alternatives will be obtained. The connection with choice rules as defined in the previous section will be stated. Then, in Section 3, a comparison among our approach and the PBA and CBA approaches will be given.

In our proposed approach, the robotic agent knowledge is represented using Defeasible Logic Programming (DeLP) [1]. A DeLP-program  $\mathcal{P}$  is denoted  $(\Pi, \Delta)$ , where  $\Pi = \Pi_f \cup \Pi_r$ , distinguishing the subsets  $\Pi_f$  of facts,  $\Pi_r$  of strict rules and the subset  $\Delta$  of defeasible rules. *Facts* are ground literals representing atomic information or the negation of atomic information. The set with all the alternatives available to the decision maker (agent) is denoted as  $X$  and in our case,  $\Pi_f = X \cup \Phi$ , where the facts in  $\Phi$  will represent the agent's perception. For instance, in Figure 1(a) a particular situation of our experimental environment is shown. Figure 1(b) shows the set  $X$  of available alternatives and (c) shows the set  $\Phi$  that represents the robot's perceptions of this particular situation.

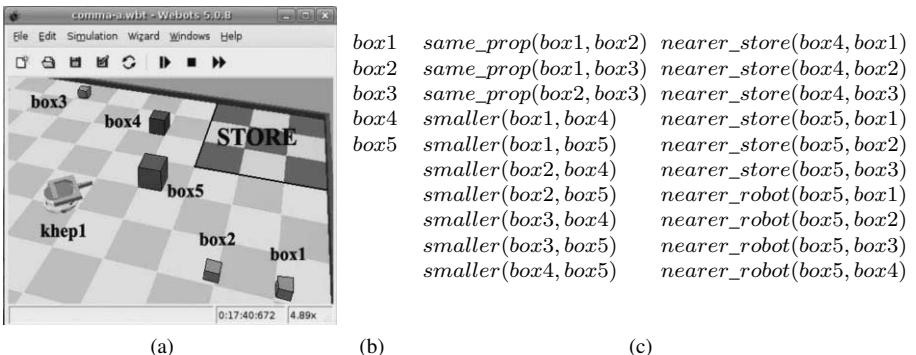


Figure 1. (a) Experimental environment, (b) Alternatives  $X$ , (c) Perceptions  $\Phi$

*Strict rules* are denoted  $L_0 \leftarrow L_1, \dots, L_n$  and represent firm information, whereas *defeasible rules* are denoted  $L_0 \leftarrow L_1, \dots, L_n$  and represent tentative information. In both cases, the *head*  $L_0$  is a literal and the *body*  $\{L_i\}_{i>0}$  is a set of literals. For example, the strict rule  $\sim better(box1, box2) \leftarrow same\_prop(box1, box2)$  states that if boxes  $box1$  and  $box2$  have the same properties, then  $box1$  is not a better alternative than  $box2$ . On the other hand, the defeasible rule  $better(box2, box4) \leftarrow smaller(box2, box4)$  expresses a reason for considering  $box2$  better than  $box4$  because is smaller.

Strict and defeasible rules are ground, however, following the usual convention [7], some examples will use “schematic rules” with variables. Figure 2 shows the defeasible and strict rules that the robot will use to compare any pair of alternatives. Observe that strong negation (“ $\sim$ ”) is allowed in the head of program rules, and hence may be used to represent contradictory knowledge.

$$\begin{aligned}
 & \text{better}(\text{Box}, \text{Obox}) \leftarrow \text{nearer\_robot}(\text{Box}, \text{Obox}) & (1) \\
 & \text{better}(\text{Box}, \text{Obox}) \leftarrow \text{nearer\_store}(\text{Box}, \text{Obox}) & (2) \\
 & \text{better}(\text{Box}, \text{Obox}) \leftarrow \text{smaller}(\text{Box}, \text{Obox}) & (3) \\
 & \sim\text{better}(\text{Box}, \text{Obox}) \leftarrow \text{nearer\_robot}(\text{Obox}, \text{Box}) & (4) \\
 & \sim\text{better}(\text{Box}, \text{Obox}) \leftarrow \text{nearer\_store}(\text{Obox}, \text{Box}) & (5) \\
 & \sim\text{better}(\text{Box}, \text{Obox}) \leftarrow \text{smaller}(\text{Obox}, \text{Box}) & (6) \\
 & \sim\text{better}(\text{Box}, \text{Obox}) \leftarrow \text{same\_prop}(\text{Obox}, \text{Box}) & (7) \\
 & \sim\text{better}(\text{Box}, \text{Obox}) \leftarrow \text{same\_prop}(\text{Box}, \text{Obox}) & (8)
 \end{aligned}$$

**Figure 2.**  $\mathcal{P}_k = (\Pi_k, \Delta_k)$

In DeLP, to deal with contradictory and dynamic information, *arguments* for conflicting pieces of information are built and then compared to decide which one *prevails*. An *argument* for a literal  $L$ , denoted  $\langle \mathcal{A}, L \rangle$ , is a minimal set of defeasible rules  $\mathcal{A} \subseteq \Delta$ , such that  $\mathcal{A} \cup \Pi$  is non-contradictory and there is a derivation for  $L$  from  $\mathcal{A} \cup \Pi$ . To establish if  $\langle \mathcal{A}, L \rangle$  is a non-defeated argument, *argument rebuttals* or *counter-arguments* that could be *defeaters* for  $\langle \mathcal{A}, L \rangle$  are considered, *i.e.*, counter-arguments that by some criterion are preferred to  $\langle \mathcal{A}, L \rangle$ . Since counter-arguments are arguments, defeaters for them may exist, and defeaters for these defeaters, and so on. Thus, a sequence of arguments called *argumentation line* is constructed, where each argument defeats its predecessor in the line (for a detailed explanation of this dialectical process see [1]).<sup>3</sup> The prevailing argument provides a warrant for the information it supports. A literal  $L$  is *warranted* from  $(\Pi, \Delta)$  if a non-defeated argument  $\mathcal{A}$  supporting  $L$  exists. Given a query  $Q$  there are four possible answers: YES, if  $Q$  is warranted; NO, if the complement of  $Q$  is warranted; UNDECIDED, if neither  $Q$  nor its complement are warranted; and UNKNOWN, if  $Q$  is not in the language of the program.

In DeLP, *generalized specificity* is used as default criterion to compare conflicting arguments, but an advantageous feature of DeLP is that the comparison criterion among arguments can be replaced in a modular way. Therefore, in our proposal, we use an appropriate literal-based criterion.<sup>4</sup> In a program  $\mathcal{P}$ , a subset of literals  $\text{comp-lits}(\mathcal{P}) \subseteq \Pi$  called *comparison literals* will be distinguished. This set of comparison literals will be used by the comparison criterion defined below.

**Definition 3 (L-order)** Let  $\mathcal{P}$  be a DeLP-program and  $\text{comp-lits}(\mathcal{P})$  the set of comparison literals in  $\mathcal{P}$ . An L-order over  $\mathcal{P}$  is a partial order<sup>5</sup> over the elements of  $\text{comp-lits}(\mathcal{P})$ .

<sup>3</sup>The implementation (interpreter) of DeLP that satisfies the semantics described in [1] is currently accessible online at <http://lidia.cs.uns.edu.ar/DeLP>.

<sup>4</sup>This comparison criterion arose as a domain-dependent criterion, but can be easily generalized to other domains.

<sup>5</sup>In the particular case that all the comparison literals are related among each other, then the *L-order* will be a total order over the elements of  $\text{comp-lits}(\mathcal{P})$ .

As stated in the introductory section, the robot will *prefer its nearer boxes, then the boxes nearer to the store, and finally the smaller ones*. These preferences will be explicitly established using a preference order among the literals *smaller*, *nearer\_store* and *nearer\_robot*. As will be shown below, an L-order must be provided as a set of facts within the program. These facts are written as  $L_1 > L_2$ , stating that a literal  $L_1$  is preferred to a literal  $L_2$ , and they will be used to decide when an argument is better than another. In particular, the L-order defined in Figure 3 represents the robot's preferences over boxes, i.e., it shows the L-order defined over program  $\mathcal{P}_k$  (Figure 2), used in our application examples. Based on a given L-order, the following argument comparison criterion can be defined.

**Definition 4 (Literal-based comparison criterion)** Let  $\mathcal{P} = (\Pi, \Delta)$  be a DeLP-program and let “ $>$ ” be an L-order over  $\mathcal{P}$ . Given two argument structures  $\langle \mathcal{A}_1, h_1 \rangle$  and  $\langle \mathcal{A}_2, h_2 \rangle$ , the argument  $\langle \mathcal{A}_1, h_1 \rangle$  will be preferred over  $\langle \mathcal{A}_2, h_2 \rangle$  iff:

1. there are two literals  $L_1$  and  $L_2$  such that  $L_1 \in^* \mathcal{A}_1$ ,  $L_2 \in^* \mathcal{A}_2$ ,  $L_1 > L_2$ , and
2. there are no literals  $L'_1$  and  $L'_2$  such that  $L'_1 \in^* \mathcal{A}_1$ ,  $L'_2 \in^* \mathcal{A}_2$ , and  $L'_2 > L'_1$ .

**Notation:**  $L \in^* \mathcal{A}$  iff there exists a defeasible rule  $(L_0 \leftarrow L_1, L_2, \dots, L_n)$  in  $\mathcal{A}$  and  $L = L_i$  for some  $i$  ( $0 \leq i \leq n$ ).

$$\text{nearer\_robot}(Z, W) > \text{nearer\_store}(W, Y) \quad (9)$$

$$\text{nearer\_robot}(Z, Y) > \text{nearer\_store}(W, Z) \quad (10)$$

$$\text{nearer\_robot}(Z, W) > \text{smaller}(W, Y) \quad (11)$$

$$\text{nearer\_robot}(Z, Y) > \text{smaller}(W, Z) \quad (12)$$

$$\text{nearer\_store}(Z, W) > \text{smaller}(W, Y) \quad (13)$$

$$\text{nearer\_store}(Z, W) > \text{smaller}(Y, Z) \quad (14)$$

$$\text{smaller}(Z, W) > \text{smaller}(W, Y) \quad (15)$$

Figure 3. L-order over *comp-lits*( $\mathcal{P}_k$ )

As above-mentioned, when an agent has to make a decision it is faced with a set of possible alternatives ( $B$ ). In our approach, the agent's decision making policy will be implemented using decision rules based on information available in  $(\Pi, \Delta)$ . A decision rule is denoted  $(D \xrightarrow{B} P, \text{not } T)$ , where  $P$  mentions the literals that *must* be warranted,  $T$  mentions the literals that *must not* be warranted,  $B$  represents the set of alternatives posed to the decision maker, and  $D$  denotes those alternatives that this rule decide to adopt from  $B$ . As we will define next, a decision rule will be used to specify under what conditions a set of decisions could be made.

**Definition 5 (Decision rule)** Let  $X$  be the set of all possible alternatives and  $B \subseteq X$  a set of alternatives (or choice experiment) that can be conceivably posed to the decision maker. A decision rule is denoted  $(D \xrightarrow{B} P, \text{not } T)$ , where  $D \subseteq B$ , the set  $P$  contains literals representing preconditions, and the set  $T$  contains literals representing constraints. Decision rules will also be denoted  $\{d_1, \dots, d_i\} \xrightarrow{B} \{p_1, \dots, p_n\}$ , not  $\{t_1, \dots, t_m\}$ .

Observe that decision rules do not belong to the DeLP-program of the agent. Therefore, we will denote with  $\Gamma_B$  the set of all available decisions rules of an agent, when

posed the choice experiment  $B$ . Decision rules are ground, but following the common convention [7], in Figure 4 we present “schematic rules” with variables. For example, rule (16) states that an alternative  $W \in B$  will be chosen, if  $W$  is better than another alternative  $Y$  and there is not a better alternative ( $Z$ ) than  $W$ . Besides, rule (17) says that two alternatives  $W, Y \in B$  with the same properties will be chosen if there is not a better alternative ( $Z$ ) than  $W$  and  $Y$ .

$$\{W\} \xleftarrow{B} \{\text{better}(W, Y)\}, \text{not } \{\text{better}(Z, W)\} \quad (16)$$

$$\{W, Y\} \xleftarrow{B} \{\text{same\_prop}(W, Y)\}, \text{not } \{\text{better}(Z, W)\} \quad (17)$$

**Figure 4.** Set of decision rules available to the robot ( $\Gamma_B$ )

**Definition 6 (Applicable decision rule)** Let  $(\Pi, \Delta)$  be a DeLP-program representing the agent’s knowledge base. Let  $B$  be the choice experiment posed to the agent and let  $\Gamma_B$  be the set of all its available decision rules. A rule  $(D \xleftarrow{B} P, \text{not } T) \in \Gamma_B$  is applicable with respect to  $(\Pi, \Delta)$ , if every precondition  $p_i \in P$  is warranted from  $(\Pi, \Delta)$  and every constraint  $t_j \in T$  fails to be warranted from  $(\Pi, \Delta)$ .

**Example 2** Consider the robot *khep1* in the scenario depicted in Figure 1(a) and the preferences shown in Figure 3. As in Example 1, the set  $\mathcal{B}$  of choice experiments will be composed by  $B_{1.1} = \{\text{box}_3, \text{box}_4, \text{box}_5\}$  and  $B_{1.2} = \{\text{box}_1, \text{box}_2, \text{box}_3, \text{box}_4, \text{box}_5\}$ . Recall from Example 1 that  $C(B_{1.1}) = \{\text{box}_5\}$ , (because despite being a big box, is near to the store and is also the only box near to the robot) and that  $C(B_{1.2}) = \{\text{box}_5\}$ .

Considering the choice experiment  $B_{1.1}$ , as boxes  $\text{box}_3$ ,  $\text{box}_4$  and  $\text{box}_5$  do not have the same properties then rule (17) cannot be applied. Besides, two non-defeated arguments ( $\mathcal{A}_1$  and  $\mathcal{A}_2$ ) supporting  $\text{better}(\text{box}_5, \text{box}_3)$  and  $\text{better}(\text{box}_5, \text{box}_4)$  (respectively) exist. Furthermore, there is no warrant for the constraint of rule (16), therefore it is applicable, being  $\text{box}_5$  the alternative selected as specified by  $C(B_{1.1})$ .

$$\begin{aligned} \mathcal{A}_1 &= \{\text{better}(\text{box}_5, \text{box}_3) \prec \text{nearer\_store}(\text{box}_5, \text{box}_3)\} \\ \mathcal{A}_2 &= \{\text{better}(\text{box}_5, \text{box}_4) \prec \text{nearer\_robot}(\text{box}_5, \text{box}_4)\} \end{aligned}$$

When considering the choice experiment  $B_{1.2}$  it holds that the boxes  $\text{box}_1$ ,  $\text{box}_2$  and  $\text{box}_3$  have the same properties. Despite this fact, rule (17) cannot be applied because its constraint is warranted, since there are two better boxes ( $\text{box}_4$  and  $\text{box}_5$ ) than  $\text{box}_1$ ,  $\text{box}_2$  and  $\text{box}_3$ . Alternatively, rule (16) can be applied because still holds that  $\text{box}_5$  is the best box, and the above-mentioned arguments  $\mathcal{A}_1$  and  $\mathcal{A}_2$  will be built by DeLP. Moreover, there will not be an argument able of warranting the constraint of decision rule (16), therefore,  $\text{box}_5$  will be the alternative selected as specified by  $C(B_{1.2})$ .

**Definition 7 (Acceptable alternatives)** Let  $B \subseteq X$  be a set of alternatives posed to the agent, let  $(\Pi, \Delta)$  be a DeLP-program representing the agent’s knowledge base, and let  $\Gamma_B$  be the agent’s set of available decision rules. Let  $\{D_i \xleftarrow{B} P_i, \text{not } T_i\}_{i=1\dots n} \subseteq \Gamma_B$  be the set of applicable decision rules with respect to  $(\Pi, \Delta)$ . The set of acceptable alternatives of the agent will be  $\Omega_B = \bigcup_{i=1}^n D_i$ .

Observe that  $\Omega_B$  implements a choice rule  $C(B)$  as introduced in Section 1 for the choice-based approach (CBA) defined in [2]. As stated for  $C(B)$ , the set  $\Omega_B$  is a subset

of  $B$  and if  $\Omega_B$  contains a single element, that element is the individual's choice from among the alternatives in  $B$ . However, if  $\Omega_B$  contains more than one element, then they represent acceptable alternatives that the agent might choose. For instance, in Example 2  $\Omega_{B_{1.1}} = \Omega_{B_{1.2}} = \{box5\}$ .

### 2.1. Proposed methodology

In the previous section the main components involved in the agent's decision making policy were presented. In order to facilitate the development of these components some guidelines are proposed below. At this end, we begin our description of the methodology making explicit some necessary assumptions:

**Assumption 1:** There must be at least one preference criterion among the alternatives.

For example, in our domain we have three comparison criteria: "proximity to the robot, proximity to the store and the boxes' size".

**Assumption 2:** For each preference criterion there must be a comparison literal belonging to the *L-order* defined in Definition 3. Each comparison literal is represented by a binary functor that states the preference between two alternatives according to the preference criterion it represents (see Figure 3). In the sequel, we will assume that all the comparison literals belong to the corresponding *L-order*.

Now we present the main methodological steps for programming the agent's knowledge:

**Step 1:** For each comparison literal (*e.g.*,  $smaller(W, Y)$ ) we add to the knowledge base two defeasible rules, one that states that alternative  $W$  is better than alternative  $Y$ ; and another rule that states that alternative  $W$  is not better than alternative  $Y$  with respect to the comparison literal. For instance, the comparison literal *smaller* has its respective rules (3) and (6), shown in Figure 2.

**Step 2:** A fact  $same\_prop(W, Y)$  should be provided in  $\Phi$ , if two alternatives  $W$  and  $Y$  have the same properties.

**Step 3:** Two strict rules have to be included in the knowledge base to account for the case of alternatives with the same properties (see rules (7) and (8) of Figure 2). This pair of rules states that if there are two alternatives with the same properties, then no one is better than the other. Since the rules are strict then no argument will defeat their conclusions.

**Step 4:** Two decision rules like the ones presented in the Figure 4 must be provided.

**Remark 4.1:** When there is more than one available alternative, it may happen that there is an alternative that is the best, or there are various alternatives with the same properties which are the best. Thus, in Figure 4 we can observe that rule (16) captures the cases where there is only one best alternative, while rule (17) is activated when there are two or more boxes with the same properties, and they are the best.

In the sequel, a DeLP-program  $(\Pi, \Delta)$  built following this methodology will be denoted as  $(\Pi, \Delta)^*$ . Therefore, based on this particular DeLP-program and on a set of decision rules, the definition of decision framework can be introduced.

**Definition 8 (Decision Framework)** Let  $X$  be the set of all possible alternatives the agent has, let  $(\Pi, \Delta)^*$  be a DeLP-program representing the agent's knowledge base and let  $\Gamma_X$  be the set of available decision rules of the agent. A decision framework will be denoted with the tuple  $\langle (\Pi, \Delta)^*, \Gamma_X \rangle$ .

### 3. Formal Comparison

In this section, we present a comparison among the approach introduced in Section 2 and the PBA and CBA approaches described in Section 1.

First, we will show what happens in the particular case that all the comparison literals belonging to the *L-order* are related among each other (*i.e.*, when *L-order* is a total order over the elements of  $\text{comp-lits}(\mathcal{P})$ ). In this particular case, following the above-proposed methodology, the set of decision rules of Figure 4 and the knowledge base of Figure 2 implement a *rational preference relation* (Definition 1). The term “implements” means that the preference relation implicitly determined by the predicate *better* is rational, or alternatively, it can be interpreted as saying that the choice behavior generated by the decision rules is coincident with the optimal choices according to this rational preference relation. This issue is formalized below.

**Theorem 1** *Let  $n$  ( $n \geq 1$ ) be the number of preference criteria that the agent has, and let  $X$  be the set of all the alternatives. Let  $\langle (\Pi, \Delta)^*, \Gamma_X \rangle$  be a decision framework, where the L-order over  $(\Pi, \Delta)^*$  is a total order. Then,  $\langle (\Pi, \Delta)^*, \Gamma_X \rangle$  implements a rational preference relation.*

(Proof is omitted due to space restrictions.)

In those cases that the agent has incomplete information about its preferences, but is possible to make a decision, it would be desirable that the agent still exhibits consistency in its decision. In particular, the decisions made by the agent will satisfy the WARP principle (see Definition 2). These issues will be formalized in Lemma 1 and Theorem 2.

When we say incomplete information about the agent’s preferences we mean that the *L-order* is in fact a partial order, and some of the comparison criteria are not related among each other. For example, in our robotic domain, if the robot is provided with two criteria and they are not related among each other, then it will not be able of deciding for any box, unless all the boxes have the same attributes values, which in fact would imply that all the boxes represent the same alternative because all their attributes coincide.

It is important to note that at least three criteria are needed to analyze the impact of incomplete information about the agent’s preferences. This is due to the fact that having three preference criteria  $pc1$ ,  $pc2$  and  $pc3$  allows us to model the two general cases that might occur with more than three criteria. These cases would be: (a) It is not specified the priority between a pair of criteria, but the other criteria are related among each other, *e.g.*, to leave unrelated  $pc1$  and  $pc2$  but is still specified the existing preference order among  $pc1$  and  $pc3$ , and  $pc2$  and  $pc3$ ; (b) It is not specified the priority between a particular criterion and the other criteria, *e.g.*, to leave  $pc1$  unrelated with respect to the other criteria, but the other criteria are related among each other (*i.e.*,  $pc2$  and  $pc3$ ).

**Lemma 1** *Let  $n$  ( $n \geq 3$ ) be the number of preference criteria that the agent has. Let  $X$  be the set of all the alternatives. Let  $B_1, B_2 (\subseteq X)$  be choice experiments presented to the agent, where  $x, y \in B_1$  and  $B_1 \subset B_2$ . Let  $\Omega_{B_1}$  and  $\Omega_{B_2}$  be the sets of acceptable alternatives of the agent with respect to  $\Gamma_{B_1}$  and  $\Gamma_{B_2}$ , correspondingly. Let  $(\Pi, \Delta)^*$  be the knowledge base of the agent. If  $\Omega_{B_1} = \{x\}$  then it holds that  $y \notin \Omega_{B_2}$ .*

(Proof is omitted due to space restrictions.)

Lemma 1 is used in the proof of Theorem 2 to show that when comparison literals that are not related among each other exist, our defeasible decision making approach still exhibits consistency in the decisions made by the agent because it satisfies the WARP principle (see Definition 2).

**Theorem 2** *Let  $n$  ( $n \geq 3$ ) be the number of preference criteria that the agent has. Let  $X$  be the set of all the alternatives. Let  $\mathcal{B} = \{B_1, \dots, B_m\}$  be a set of nonempty subsets of  $X$ ; that is, every element of  $\mathcal{B}$  is a set  $B_i \subseteq X$  representing a choice experiment presented to the agent. Let  $\Omega_{B_i}$  be the set of acceptable alternatives of the agent with respect to  $\Gamma_{B_i}$ . Let  $(\Pi, \Delta)^*$  be a DeLP-program representing the knowledge base of the agent. Then the decision framework  $\langle (\Pi, \Delta)^*, \Gamma_{\mathcal{B}} \rangle$  implements a choice structure  $(\mathcal{B}, C(\cdot))$  that satisfies the weak axiom of revealed preference.*

(Proof is omitted due to space restrictions.)

#### 4. Related work and Conclusions

Our approach to decision making is related to other works which use argumentative processes as a fundamental component in the decision making of an agent. For instance, in [8] it is highlighted the fundamental role of argumentation for the management of uncertainty in symbolic decision making, where several applications based on argumentative approaches are presented as empirical evidence of this claim. In [9], an agent called *Drama* incorporates an argumentation component which provides the ability to make flexible and context dependent decisions about medical treatment, based on several information sources (perspectives). The influence of different contexts that arise in changing environments is also considered by Kakas *et al.* in [10] where an argumentation-based framework supports the decision making of an agent modular architecture. In this case, arguments and their strength depend on the particular context that the agent finds himself.

Some recent works [11,12] use argumentation as a powerful tool for explaining qualitative decision making under uncertainty. In this case, the proposal aims to explain optimistic and pessimistic decision criteria in terms of arguments in favor or against each alternative. Bonet and Geffner [13] also propose a qualitative model of decision making where *reasons* for and against decisions interact and play a fundamental role in the decision procedure.

Respect to the traditional approaches to classical decision theory, our work mainly differs in that the analysis is directly addressed on the agent's preference relation and not on a utility function that represents this relation, as usual in these cases. This aspect allows us to establish a direct connection between our argumentation-based decision making approach and more essentials approaches for modeling individual choice behavior.

In that sense, we show that our approach can guarantee the rationality property of the preference relation implicitly implemented, when all the relevant information about the preference criteria are specified. On the other hand, when this information is not fully specified our approach can still guarantee some degree of coherence according to the WARP principle and the choice rules approach to decision making.

These properties of our approach can only be achieved if the designer of the agent's decision components follows some methodological steps. At this end, our approach includes a simple methodology which exactly characterizes the DeLP-program representing the agent's knowledge base used in the definition of the agent's decision framework.

Another interesting aspect of our proposal is that the argument comparison criteria used for determining if an alternative is better than another during the argumentation process are explicitly stated in the DeLP-program. In that way, these criteria can be easily modified, added or removed and therefore we can change the agent's decision policy in a flexible way with minor changes in the agent's knowledge base.

A first extension of our work would be to consider the presence of multiple agents in the environment. In this new scenario, formalisms related to agents coordination need to be considered and the connections of our approach to these mechanisms also need to be stated.

## Acknowledgements

This work is partially supported by Universidad Nacional de San Luis, Universidad Nacional del Sur, ANPCyT and CONICET (PIP 5050).

## References

- [1] Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming: an argumentative approach. *Theory and Practice of Logic Programming*, 4(2):95–138, 2004.
- [2] Andreu Mas-Collel, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, 1995.
- [3] K-Team. Khepera 2. <http://www.k-team.com>, 2002.
- [4] Olivier Michel. Webots: Professional mobile robot simulation. *Journal of Advanced Robotics Systems*, 1(1):39–42, 2004.
- [5] Edgardo Ferretti, Marcelo Errecalde, Alejandro García, and Guillermo Simari. Khedelp: A framework to support defeasible logic programming for the khepera robots. In *International Symposium on Robotics and Automation (ISRA)*, pages 98–103, San Miguel Regla, Hidalgo, México, August 2006.
- [6] Edgardo Ferretti, Marcelo Luis Errecalde, Alejandro Javier García, and Guillermo Ricardo Simari. Khepera robots with argumentative reasoning. In *Proceedings of the 4th International AMIRE Symposium*, pages 199–206, Buenos Aires, Argentina, October 2007.
- [7] V. Lifschitz. Foundations of logic programming. In Gerhard Brewka, editor, *Principles of Knowledge Representation*, pages 69–127. CSLI Publications, Stanford, California, 1996.
- [8] Simon Parsons and John Fox. Argumentation and decision making: A position paper. In *FAPR '96: Proceedings of the International Conference on Formal and Applied Practical Reasoning*, pages 705–709, London, UK, 1996. Springer-Verlag.
- [9] Katie Atkinson, Trevor J. M. Bench-Capon, and Sanjay Modgil. Argumentation for decision support. In S. Bressan, J. Küng, and R. Wagner, editors, *17th International Conference on Database and Expert Systems Applications (DEXA)*, volume 4080 of *LNCS*, pages 822–831, Berlin, 2006. Springer.
- [10] Antonis Kakas and Pavlos Moraitsis. Argumentation based decision making for autonomous agents. In *AAMAS '03*, pages 883–890, New York, NY, USA, 2003. ACM.
- [11] Leila Amgoud and Henri Prade. Using arguments for making decisions: a possibilistic logic approach. In *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 10–17, Arlington, Virginia, United States, 2004. AUAI Press.
- [12] Leila Amgoud and Henri Prade. Explaining qualitative decision under uncertainty by argumentation. In *The Twenty-first National Conference on Artificial Intelligence (AAAI-06)*, 2006.
- [13] Blai Bonet and Hector Geffner. Arguing for decisions: A qualitative model of decision making. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence*, pages 98–105, 1996.

# Hybrid argumentation and its properties

Dorian GAERTNER and Francesca TONI

*Department of Computing, Imperial College London, UK*

**Abstract.** We present a variant of AB-dispute derivations for assumption-based argumentation (ABA), that can be used for determining the admissibility of claims. ABA reduces the problem of computing arguments to the problem of computing assumptions supporting these arguments. Whereas the original AB-dispute derivations only manipulate sets of assumptions, our variant also renders explicit the underlying dialectical structure of arguments (by a proponent) and counter-arguments (by an opponent), and thus supports a hybrid of ABA and abstract argumentation beneficial to developing applications of argumentation where explicit justifications of claims in terms of full dialectical structures are required. We prove that the proposed variant of AB-dispute derivations is correct.

**Keywords.** Abstract Argumentation, Assumption-based Argumentation, Computation

## 1. Introduction

Argumentation has proved to be a useful abstraction mechanism for understanding several problems, and a number of computational frameworks for argumentation have been proposed in order to provide tools to address these problems. These frameworks are mostly based upon abstract argumentation [6], that focuses on determining the admissibility of arguments based upon their capability to counter-attack all arguments attacking them, while being conflict-free. In abstract argumentation the *arguments* and the *attack* relation between arguments are seen as primitive notions. The abstract view of argumentation is equipped with intuitive and simple computational models (e.g. [4,14]), but does not allow to address the problems of (i) how to find arguments, (ii) how to determine attacks and (iii) how to exploit the fact that different arguments may share premises. Assumption-based argumentation [1,7,9] is a general-purpose framework for argumentation, where arguments, rather than being a primitive concept, are defined as *backward deductions* (using sets of *rules* in an underlying logic) supported by sets of *assumptions*, and the notion of attack amongst arguments is reduced to that of *contrary* of assumptions. Intuitively, assumptions are sentences that can be assumed to hold but can be questioned and disputed (as opposed to axioms that are instead beyond dispute), and the contrary of an assumption stands for the reason why that assumption may be undermined and thus may need to be dropped.

Existing computational models for assumption-based argumentation [7,9] allow to determine the “acceptability” of claims under the semantics of credulous, admissible extensions as well as under two sceptical semantics (of grounded and ideal extensions). In this paper, we focus on the computational model for admissibility, called AB-dispute derivations [7,9]. These derivations can be seen as a game between two (fictional) players

– a *proponent* and an *opponent* – with rules roughly as follows: the opponent can dispute the proponent’s arguments by attacking one of the arguments’ supporting assumptions; the proponent can in turn defend its arguments by counter-attacking the opponent’s attacks with other arguments, possibly with the aid of other defending assumptions; the proponent does not need to counter-attack any assumption it has already attacked previously or defend any assumption it has already defended previously; the proponent cannot attack any of its own assumptions. While conducting this game, the players explore implicitly a dialectical structure of arguments by the proponent, counter-arguments by the opponent, arguments by the proponent attacking the counter-arguments and so on. However, while doing so, AB-dispute derivations only keep track of the assumptions underlying these arguments, and the dialectical structure is lost.

In this paper, we define generalised structured AB-dispute derivations, a variant of AB-dispute derivations computing explicitly the dialectical structure hidden in AB-dispute derivations and thus providing a hybrid ABA-abstract argumentation mechanisms. Our structured AB-dispute derivations are defined for a generalisation of ABA allowing for *multiple contraries*. Also, they are a (non-trivial) generalisation of the structured AB-dispute derivations defined in [11]: whereas those relied upon a special *patient* selection function for exploring and building arguments, we do not commit to any selection function (or other design choice). Moreover, whereas in [11] we were concerned with the implementation of structured AB-dispute derivations, here we are concerned with formal proofs of correctness (missing in [11]).

The paper is organised as follows: in Section 2 we present the background on ABA and existing notions of AB-dispute derivations. In Section 3 we detail our novel generalised structured AB-dispute derivations. We prove correctness results in Section 4 and conclude in Section 5.

## 2. Assumption-based argumentation

This section provides the basic background on assumption-based argumentation (ABA), see [1, 7, 9] for details. An ABA framework is a tuple  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{x}} \rangle$  where

- $(\mathcal{L}, \mathcal{R})$  is a *deductive system*, consisting of a language  $\mathcal{L}$  and a set  $\mathcal{R}$  of inference rules,
- $\mathcal{A} \subseteq \mathcal{L}$ , referred to as the set of *assumptions*,
- $\overline{\phantom{x}}$  is a (total) mapping from  $\mathcal{A}$  into  $\mathcal{L}$ , where  $\overline{x}$  is referred to as the *contrary* of  $x$ .

We will assume that the inference rules in  $\mathcal{R}$  have the syntax  $l_0 \leftarrow l_1, \dots, l_n$  (for  $n \geq 0$ ) where  $l_i \in \mathcal{L}$ . We will refer to  $l_0$  and  $l_1, \dots, l_n$  as the *head* and the *body* of the rule, respectively. We will represent the rule  $l \leftarrow \dots$  simply as  $l$ . As in [7], we will restrict attention to *flat* ABA frameworks, such that if  $l \in \mathcal{A}$ , then there exists no inference rule of the form  $l \leftarrow l_1, \dots, l_n \in \mathcal{R}$ , for any  $n \geq 0$ .

We will adopt a generalisation of ABA frameworks, first given in [10], whereby assumptions allow *multiple contraries* (i.e.  $\overline{\phantom{x}}$  is a (total) mapping from  $\mathcal{A}$  into  $\wp(\mathcal{L})$ ). As argued in [10], multiple contraries are a useful generalisation to ease representation and comprehension of ABA frameworks. However, they do not really extend the expressive power of ABA frameworks. Indeed, there is a one-to-one correspondence between original ABA frameworks and ABA frameworks with multiple contraries. For example, consider the framework with multiple contraries  $\langle \mathcal{L}_g, \mathcal{R}_g, \mathcal{A}_g, \overline{\phantom{x}} \rangle$  where:

$$\mathcal{L}_g = \{p, a, c_1, b, c_3, c_4\}, \quad \mathcal{R}_g = \{p \leftarrow a; c_1 \leftarrow b; c_3\}, \quad \mathcal{A}_g = \{a, b\}, \\ \bar{a} = \{c_1, c_2, c_3\} \text{ and } \bar{b} = \{c_4\}$$

This can be turned into a corresponding original framework  $\langle \mathcal{L}_o, \mathcal{R}_o, \mathcal{A}_o, \overline{\cdot} \rangle$  where:

$$\mathcal{L}_o = \mathcal{L}_g \cup \{aux\}, \quad \mathcal{R}_o = \mathcal{R}_g \cup \{aux \leftarrow c_1; aux \leftarrow c_2; aux \leftarrow c_3\}, \quad \mathcal{A}_o = \mathcal{A}_g, \\ \bar{a} = aux \text{ and } \bar{b} = c_4$$

We will use this correspondence to simplify some of the proofs in Section 4.

Given an ABA framework, an *argument* in favour of a sentence  $x \in \mathcal{L}$  supported by a set of assumptions  $X$ , denoted  $X \vdash x$ , is a (backward) deduction from  $x$  to  $X$ , obtained by applying backwards the rules in  $\mathcal{R}$ . For example, given  $\langle \mathcal{L}_g, \mathcal{R}_g, \mathcal{A}_g, \overline{\cdot} \rangle$  above,  $\{a\} \vdash p$  is an argument.

In order to determine whether a conclusion (set of sentences) should be drawn, a set of assumptions needs to be identified providing an “acceptable” support for the conclusion. Various notions of “acceptable” support can be formalised, using a notion of “attack” amongst sets of assumptions whereby  $X_1$  attacks  $X_2$  iff there is an argument in favour of some  $y \in \bar{x}$  supported by (a subset of)  $X_1$  where  $x$  is in  $X_2$  (for example, given  $\langle \mathcal{L}_g, \mathcal{R}_g, \mathcal{A}_g, \overline{\cdot} \rangle$  above,  $\{b\} \vdash c_1$  attacks  $\{a\} \vdash p$ ). In this paper, we will consider the following notions of “acceptable” set of assumptions and support:

- a set of assumptions is *admissible* iff it does not attack itself and it attacks every set of assumptions attacking it;
- an *admissible support* for a claim is an admissible set  $X$  of assumptions such that there exists an argument in favour of the claim supported by a subset of  $X$ .

As shown in [9], there is a one-to-one correspondence between admissible supports for conclusions, in terms of sets of assumptions, and admissible sets of arguments (supported by assumptions), in the sense of [6].

AB-dispute derivations [7,9] allow to compute admissible supports for given claims (if any exists). They are finite sequences of tuples  $\langle \mathcal{P}_i, \mathcal{O}_i, D_i, C_i \rangle$  where  $\mathcal{P}_i$  and  $\mathcal{O}_i$  represent (the set of sentences and the set of sets of sentences held by) the proponent and opponent (respectively) at step  $i$  in the dispute,  $D_i$  is the set of assumptions generated by the proponent in support of the initial claim and to defend itself against the opponent, and  $C_i$  is the set of assumptions in counter-arguments generated by the opponent that the proponent has chosen as “culprits” to counter-attack. The tuple at the start of a derivation for claim  $x$  is:  $\langle \{x\}, \{\}, \mathcal{A} \cap \{x\}, \{\} \rangle$ .

**Example 1** The AB-dispute derivation for the framework ( $\mathcal{L}$  is omitted for brevity)

$$\mathcal{R} = \{p \leftarrow a, r; c_1 \leftarrow b, s; c_1 \leftarrow t; c_2 \leftarrow q; q; r \leftarrow e\}, \quad \mathcal{A} = \{a, b, e\}, \\ \bar{a} = \{c_1\} \text{ and } \bar{b} = \{c_2\} \text{ and } \bar{e} = \{z\}$$

computes an admissible support  $\{a, e\}$  for claim  $p$  as can be seen in table 1.

AB-dispute derivations make use of a *selection function* to pick a sentence from the support of a “potential argument” to expand it further (if it is not an assumption) or to identify a possible point of attack (if it is an assumption). Such selection can be based on random choice, on whether or not the sentence is an assumption or on more complex criteria. The sentence selected by the chosen selection function at each step is underlined in the derivation in table 1. AB-dispute derivations implicitly compute a dialectical structure of (“potential”) arguments. For the derivation in table 1, these arguments are (1)

Step	Proponent	Opponent	DefenseSet	Culprits
0	<u>p</u>			
1	<u>a</u> , r		a	
2	r	{c1}	a	
3	r	{b, s}, {t}	a	
4	<u>c2</u> , r	{t}	a	b
5	<u>q</u> , r	{t}	a	b
6	<u>r</u>	{t}	a	b
7	<u>e</u>	{t}	a, e	b
8		{t}, {z}	a, e	b
9		{z}	a, e	b
10			<b>a, e</b>	<b>b</b>

Table 1. AB-dispute derivation for example 1.

$\{a, e\} \vdash p$ , (2)  $\{b, s\} \vdash c_1$ , (3)  $\{\} \vdash c_2$ , where (3) attacks (2) and (2) “potentially” attacks (1) (here, (2) is a “potential” argument as its support contains a non-assumption  $s$  - see section 3 for a formal definition). This structure however is hidden in the computation. In [11], we presented structured AB-dispute derivations computing this structure in the case of *patient* selection functions, namely always selecting non-assumptions first (the selection function used in table 1 is not patient). Below, we define *generalised structured AB-dispute derivations* working with any selection function. We also formalise many of the concepts intuitively used in [11] and prove some formal results for the generalised structured AB-dispute derivations we define here.

### 3. Generalised structured AB-dispute derivations

Our *generalised structured AB-dispute derivations*, that we refer to simply as structured AB-dispute derivations, are sequences of tuples of the form  $\langle \mathcal{P}_i, \mathcal{O}_i, D_i, C_i, A_i, R_i \rangle$ . The elements  $D_i$  and  $C_i$  are the defense set and the set of culprits, exactly as in the original AB-dispute derivations. The elements  $\mathcal{P}_i$  and  $\mathcal{O}_i$ , as before, represent the state of the proponent and opponent, but they are no longer sets of sentences and sets of sets of sentences, respectively. Instead, they consist of “potential arguments” together with information about which arguments they “potentially attack”.

**Definition 1** A potential argument  $(Y, X \vdash x)$  in favour of a sentence  $x \in \mathcal{L}$  supported by  $(Y, X)$ , with  $X \subseteq \mathcal{A}$  and  $Y \subseteq \mathcal{L}$  is a (backward) deduction from  $x$  to  $X \cup Y$ , obtained by applying backwards the rules in  $\mathcal{R}$ .

Trivially, a potential argument  $(\{\}, X \vdash x)$  corresponds to an argument  $X \vdash x$  as in conventional ABA. Below, we will refer to arguments in conventional ABA as *actual arguments*. A potential argument  $(Y, X \vdash x)$  with  $Y \subseteq \mathcal{A}$  also corresponds to an actual argument  $X \cup Y \vdash x$ .

Intuitively, potential arguments correspond to intermediate stages in the construction of actual arguments. It may be possible to turn a potential argument into zero, one, or many actual arguments, depending on whether all non-assumptions in  $Y$  can be reduced to assumptions via the backward application of rules in  $\mathcal{R}$ . In example 1,  $(\{r\}, \{a\} \vdash p)$

is a potential argument that can be turned into a single actual argument  $\{a, e\} \vdash p$  and  $(\{s\}, \{b\} \vdash c_1)$  is a potential argument that cannot be turned into an actual argument.

The notion of attack between (actual) arguments can be generalised to a notion of “potential attack” between potential arguments:

**Definition 2** A potential argument  $(Y_1, X_1 \vdash x_1)$  potentially attacks a potential argument  $(Y_2, X_2 \vdash x_2)$  iff  $x_1 \in \bar{z}$  for some  $z \in (Y_2 \cap \mathcal{A}) \cup X_2$ .

Trivially, potential attacks correspond to actual attacks whenever the potential arguments correspond to actual arguments.

The two new elements in structured AB-dispute derivations,  $A_i$  and  $R_i$ , hold, respectively, the currently computed (potential and actual) arguments and a binary relation between these arguments, corresponding to (potential and actual) attacks. In order to ease the representation of  $R_i$ , we adopt a labelling convention for arguments, namely

- $\mathcal{P}_i$  and  $\mathcal{O}_i$  are sets of expressions of the form:  $id : (Y, X \vdash x) \rightsquigarrow id^*$  indicating a (potential or actual) argument  $(Y, X \vdash x)$  labelled  $id$  (potentially or actually) attacking another argument labelled  $id^*$
- $A_i$  consists of expressions of the form  $id : (Y, X \vdash x)$  representing a (potential or actual) argument  $(Y, X \vdash x)$  labelled  $id$
- $R_i$  is a set of expressions of the form  $id \rightsquigarrow id^*$  standing for “the (potential or actual) argument labelled by  $id$  (potentially or actually) attacks the (potential or actual) argument labelled by  $id^*$ ”

Structured AB-dispute derivations interleave the construction of arguments and their evaluation (with respect to the admissibility semantics) and thus need to store potential arguments (in the components  $\mathcal{P}_i$  and  $\mathcal{O}_i$ ). Once these arguments are evaluated (with respect to admissibility) they are eliminated from  $\mathcal{P}_i$  or  $\mathcal{O}_i$  and stored in  $A_i$  (with  $R_i$  also appropriately modified).

For the purpose of labelling arguments, we will assume the existence of a function `newLabel()` that returns a fresh label every time it is invoked. We will adopt a marking mechanism for specifying our structured AB-dispute derivations, so that sentences in the support of arguments in  $\mathcal{P}_i$  and  $\mathcal{O}_i$  are marked after they have been selected (by the selection function). We will assume that the selection function will always select a sentence in the unmarked bit of the support of a potential argument. This is similar to the marking mechanism adopted for elements of  $\mathcal{O}_i$  in IB-dispute derivations in [9]. However, here we store marked assumptions in the  $X$  component of the support of a potential argument  $(Y, X \vdash x)$ , and unmarked elements in its  $Y$  component. Moreover, here the marking is needed in order to compute the dialectical structures, whereas in [9] it was needed to ensure the correctness of IB-dispute derivations.

**Definition 3** A (generalised) structured AB-dispute derivation of a defense set  $D$  and of a dialectical structure  $(A, R)$  for a sentence  $\alpha \in \mathcal{L}$  is a finite sequence of tuples  $\langle \mathcal{P}_0, \mathcal{O}_0, D_0, C_0, A_0, R_0 \rangle, \dots, \langle \mathcal{P}_i, \mathcal{O}_i, D_i, C_i, A_i, R_i \rangle, \dots, \langle \mathcal{P}_n, \mathcal{O}_n, D_n, C_n, A_n, R_n \rangle$  where initially

$\mathcal{P}_0 = \{l_1 : ([\{\alpha\}, \{\}] \vdash \alpha) \rightsquigarrow \emptyset\}$  where  $l_1 = \text{newLabel}()$  and  $\emptyset$  is a special label representing that this does not attack any other arguments

$$\begin{aligned} D_0 &= \mathcal{A} \cap \{\alpha\} \\ \mathcal{O}_0 &= C_0 = A_0 = R_0 = \{\} \end{aligned}$$

and upon termination

$$\begin{aligned} \mathcal{P}_n &= \mathcal{O}_n = \{\} \\ D &= D_n, A = A_n, R = R_n \end{aligned}$$

and for every  $0 \leq i < n$ , one potential argument  $\text{curArg}$ , of the form  $l : ([S_u, S_m] \vdash G) \rightsquigarrow l_e$ , is chosen in either  $\mathcal{P}_i$  or  $\mathcal{O}_i$ , a sentence  $\sigma$  is selected in  $S_u$ , and:

1. If  $\text{curArg}$  is chosen in  $\mathcal{P}_i$  then

- (i) if  $\sigma$  is an assumption then

$$\begin{aligned} \mathcal{P}_{i+1} &= \mathcal{P}_i - \{\text{curArg}\} \cup \text{newP} \\ \mathcal{O}_{i+1} &= \mathcal{O}_i \cup \{\text{new} : ([\{\}, \{\}] \vdash x) \rightsquigarrow l \mid x \in \bar{\sigma} \text{ and } \text{new} = \text{newLabel}(\)} \\ D_{i+1} &= D_i \\ C_{i+1} &= C_i \\ A_{i+1} &= A_i \cup \text{newArg} \\ R_{i+1} &= R_i \cup \text{newRel} \end{aligned}$$

where  $\text{newP}$ ,  $\text{newArg}$ ,  $\text{newRel}$  are defined in the table below, separating out whether or not the selected  $\sigma$  was the last unmarked element in the premise of  $\text{curArg}$  (i.e.  $S_u = \{\sigma\}$ )

$S_u - \{\sigma\} = \{\}$	$S_u - \{\sigma\} \neq \{\}$
$\text{newP} = \{\}$	$\text{newP} = \{l : ([S_u - \{\sigma\}, S_m \cup \{\sigma\}] \vdash G) \rightsquigarrow l_e\}$
$\text{newArg} = \{l : ([\{\}, S_m \cup \{\sigma\}] \vdash G)\}$	$\text{newArg} = \{\}$
$\text{newRel} = \{l \rightsquigarrow l_e\}$	$\text{newRel} = \{\}$

- (ii) if  $\sigma$  is a non-assumption and there exists some rule of the form  $\sigma \leftarrow B \in \mathcal{R}$  such that  $C_i \cap B = \{\}$  then<sup>1</sup>

$$\begin{aligned} \mathcal{P}_{i+1} &= \mathcal{P}_i - \{\text{curArg}\} \cup \text{newP} \\ \mathcal{O}_{i+1} &= \mathcal{O}_i \\ D_{i+1} &= D_i \cup (\mathcal{A} \cap B) \\ C_{i+1} &= C_i \\ A_{i+1} &= A_i \cup \text{newArg} \\ R_{i+1} &= R_i \cup \text{newRel} \end{aligned}$$

where  $\text{newP}$ ,  $\text{newArg}$ ,  $\text{newRel}$  are defined in the table below

$(S_u - \{\sigma\}) \cup (B - D_i) = \{\}$	$(S_u - \{\sigma\}) \cup (B - D_i) \neq \{\}$
$\text{newP} = \{\}$	$\text{newP} = \{l : ([S'_u, S'_m] \vdash G) \rightsquigarrow l_e\}$
$\text{newArg} = \{l : ([\{\}, S_m \cup B] \vdash G)\}$	$\text{newArg} = \{\}$
$\text{newRel} = \{l \rightsquigarrow l_e\}$	$\text{newRel} = \{\}$ where $S'_u = (S_u - \{\sigma\}) \cup (B - D_i)$ and $S'_m = S_m \cup (B \cap D_i)$

<sup>1</sup>We treat  $B$  and all bodies of inference rules in  $\mathcal{R}$  as sets.

2. If  $curArg$  is chosen from  $\mathcal{O}_i$  then

(i) if  $\sigma$  is an assumption then

(a) either  $\sigma$  is ignored, i.e.

$$\mathcal{P}_{i+1} = \mathcal{P}_i$$

$$\mathcal{O}_{i+1} = \mathcal{O}_i - \{curArg\} \cup \{l : ((S_u - \{\sigma\}), (S_m \cup \{\sigma\})) \vdash G\} \rightsquigarrow l_e\}$$

$$D_{i+1} = D_i$$

$$C_{i+1} = C_i$$

$$A_{i+1} = A_i$$

$$R_{i+1} = R_i$$

(b) or  $\sigma \in C_i$  and  $\sigma \notin D_i$

$$\mathcal{P}_{i+1} = \mathcal{P}_i$$

$$\mathcal{O}_{i+1} = \mathcal{O}_i - \{curArg\}$$

$$D_{i+1} = D_i$$

$$C_{i+1} = C_i$$

$$A_{i+1} = A_i \cup \{l : ([S_u - \{\sigma\}], S_m \cup \{\sigma\}) \vdash G\}$$

$$R_{i+1} = R_i \cup \{l \rightsquigarrow l_e\} \cup \{someLabel \rightsquigarrow l\}$$

where  $someLabel$  is any label such that, for some value of  $X, Y, Z$  and for some  $x \in \bar{\sigma}$ , either  $someLabel : ([X, Y] \vdash x) \in A_i$  or  $someLabel : ([\{X\}, \{Y\}] \vdash x) \rightsquigarrow Z \in P_i$ .<sup>2</sup>

(c) or  $\sigma \notin C_i$  and  $\sigma \notin D_i$  and

$$\mathcal{P}_{i+1} = \mathcal{P}_i \cup \{new : ([\{x\}, \{\}] \vdash x) \rightsquigarrow l\}$$

where  $x \in \bar{\sigma}$  and  $new = newLabel()$

$$\mathcal{O}_{i+1} = \mathcal{O}_i - \{curArg\}$$

$$D_{i+1} = D_i \cup (\mathcal{A} \cap \{x\})$$

$$C_{i+1} = C_i \cup \{\sigma\}$$

$$A_{i+1} = A_i \cup \{l : ([S_u - \{\sigma\}], S_m \cup \{\sigma\}) \vdash G\}$$

$$R_{i+1} = R_i \cup \{l \rightsquigarrow l_e\}$$

(ii) if  $\sigma$  is a non-assumption, then

$$\mathcal{P}_{i+1} = \mathcal{P}_i$$

$$\mathcal{O}_{i+1} = \mathcal{O}_i - \{curArg\} \cup \{l : ((S_u - \{\sigma\}) \cup B, S_m) \vdash G\} \rightsquigarrow l_e |$$

$\sigma \leftarrow B \in \mathcal{R}$  and  $B \cap C_i = \{\}$  }

$$D_{i+1} = D_i$$

$$C_{i+1} = C_i$$

$$A_{i+1} = A_i \cup \{n : ((S_u - \{\sigma\}) \cup (B - C_i), S_m \cup (B \cap C_i)) \vdash G\} |$$

$n = newLabel()$  and  $\sigma \leftarrow B \in \mathcal{R}$  and  $B \cap C_i \neq \{\}$  }

$$R_{i+1} = R_i \cup \{m \rightsquigarrow n \mid \sigma \leftarrow B \in \mathcal{R}$$
 and  $B \cap C_i \neq \{\}$  and  

$m = find\_label(B \cap C_i)$

$$\cup \{n \rightsquigarrow l_e \mid n : ([S'_u, S'_m] \vdash l) \in A_{i+1} - A_i\}$$

where  $find\_label(Set) = someLbl$  such that  $\omega \in Set$  and  
 $((someLbl : ([X, Y] \vdash \omega)) \in A_i$  or  $(someLbl : ([X, Y] \vdash \omega) \rightsquigarrow Z) \in P_i$ )

---

<sup>2</sup>If  $\sigma \in C_i$ , either the culprit  $\sigma$  is already defeated or it is on the proponent's agenda of things to be defeated. One must search through both  $P_i$  and  $A_i$  to find  $someLabel$ .

Intuitively, three choices have to be made at each step in a derivation. First a (fictional) *player* must be chosen: either the proponent ( $\mathcal{P}_i$ ) or the opponent ( $\mathcal{O}_i$ ). Next, from the chosen set, one (potential or actual) argument *curArg* needs to be chosen for further consideration. *curArg* will be of the form  $l : ([S_u, S_m] \vdash G) \rightsquigarrow l_e$ . Finally, one element  $\sigma$  from (the unmarked part  $S_u$  of) the support of *curArg* is selected. There are now four main cases to consider, depending on the player and whether  $\sigma$  is an assumption or not.

In case 1(i), the proponent plays and  $\sigma$  is an assumption: this is simply marked, and new potential arguments for the contraries of  $\sigma$  are added to the opponent. Moreover, if  $\sigma$  is the last unmarked element, the dialectical structure also gets augmented (note that in this case the argument added to  $A_i$  is an actual argument).

In case 1(ii), the proponent plays and  $\sigma$  is a non-assumption: this is unfolded using a rule with body  $B$ . If  $B$  is empty or all elements in  $B$  are assumptions that have already been defended (i.e. they are in  $D_i$ ), then the dialectical structure gets augmented (note that in this case the argument added to  $A_i$  is an actual argument). Otherwise,  $\sigma$  in *curArg* is unfolded to the rule body  $B$ : The part of  $B$  that is already in the defense set is added to the marked elements  $S_m$  (and hence treated as if already considered), whereas the part of  $B$  that is not yet in the defense set is added to  $S_u$  for future consideration.

In case 2(i), the opponent plays and  $\sigma$  is an assumption. If  $\sigma \in D_i$ , then the only option is to ignore it (case 2ia), as choosing such a  $\sigma$  as a culprit would make the defense set attack itself and hence not be admissible.

If however  $\sigma \notin D_i$ , then it could be a culprit (but note that it could also be ignored). If  $\sigma$  is already a known culprit ( $\sigma \in C_i$ ), then case 2ib applies and the potential attack *curArg* can be moved to  $A_i$  (and  $R_i$  be appropriately modified, too) since either  $\sigma$  has already been defeated or the fact that it needs to be defeated must already have been “stored” in  $\mathcal{P}_i$ . Here, the argument defeating  $\sigma$  is labelled with *someLabel*.

If  $\sigma$  is not a known culprit yet ( $\sigma \notin C_i$ ), then we add it to the set of culprits and pick one of its contraries (say,  $x$ ) for the proponent to show (in order to thereby counter-attack *curArg*). Note that, for a derivation to exist, all potential arguments in all  $\mathcal{O}_i$  need to be defeated, as otherwise the termination condition  $\mathcal{O}_n = \{\}$  would not be met. If some chosen culprit cannot be defeated, then the implementation of the algorithm can resort to backtracking on some of the choices (either the culprit itself, that can be ignored, or the contrary of the chosen culprit, or one of the rules at step 1(ii) for generating the argument attacking the culprit).

In case 2(ii), the opponent plays and  $\sigma$  is a non-assumption. Here, the opponent expands  $\sigma$  in all possible ways (i.e. using all possible rules). *curArg* is replaced with many new potential arguments, one for each applicable rule which has no known culprit in its body. For each applicable rule which has some known culprit in its body, we extend the dialectical structure by adding to  $A_i$  one potential argument for each rule that had culprits in its body.  $R_i$  is also augmented appropriately.

Let us now consider the ABA framework in example 1. A structured AB-dispute derivation exists for claim  $p$ , e.g. as given table 2 (others exist for other choices of players and selection functions, here we have used the same choices and selection functions as in table 1). Note that we obtain the same number of steps, the same defense set and the same set of culprits as for ordinary AB-dispute derivations (table 1).

#	<i>Proponent</i>	<i>Opponent</i>	D	C	A	R
0	$l_1 : ([\{p\}, \{\}] \vdash p) \rightsquigarrow \emptyset$					
1	$l_1 : ([\{a, r\}, \{\}] \vdash p) \rightsquigarrow \emptyset$		a			
2	$l_1 : ([\{r\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_2 : ([\{c_1\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a			
3	$l_1 : ([\{r\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_{2.1} : ([\{b, s\}, \{\}] \vdash c_1) \rightsquigarrow l_1,$ $l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a			
4	$l_3 : ([\{c_2\}, \{\}] \vdash c_2) \rightsquigarrow l_{2.1},$ $l_1 : ([\{r\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1)$	$l_{2.1} \rightsquigarrow l_1$
5	$l_3 : ([\{q\}, \{\}] \vdash c_2) \rightsquigarrow l_{2.1},$ $l_1 : ([\{r\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1)$	$l_{2.1} \rightsquigarrow l_1$
6	$l_1 : ([\{r\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1),$ $l_3 : ([\{\}, \{\}] \vdash c_2)$	$l_{2.1} \rightsquigarrow l_1,$ $l_3 \rightsquigarrow l_{2.1}$
7	$l_1 : ([\{e\}, \{a\}] \vdash p) \rightsquigarrow \emptyset$	$l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1$	a, e	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1),$ $l_3 : ([\{\}, \{\}] \vdash c_2)$	$l_{2.1} \rightsquigarrow l_1,$ $l_3 \rightsquigarrow l_{2.1}$
8		$l_{2.2} : ([\{t\}, \{\}] \vdash c_1) \rightsquigarrow l_1,$ $l_4 : ([\{z\}, \{\}] \vdash z) \rightsquigarrow l_1$	a, e	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1),$ $l_3 : ([\{\}, \{\}] \vdash c_2),$ $l_1 : ([\{\}, \{a, e\}] \vdash p)$	$l_{2.1} \rightsquigarrow l_1,$ $l_3 \rightsquigarrow l_{2.1},$ $l_1 \rightsquigarrow \emptyset$
9		$l_4 : ([\{z\}, \{\}] \vdash z) \rightsquigarrow l_1$	a, e	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1),$ $l_3 : ([\{\}, \{\}] \vdash c_2),$ $l_1 : ([\{\}, \{a, e\}] \vdash p)$	$l_{2.1} \rightsquigarrow l_1,$ $l_3 \rightsquigarrow l_{2.1},$ $l_1 \rightsquigarrow \emptyset$
10			a, e	b	$l_{2.1} : ([\{s\}, \{b\}] \vdash c_1),$ $l_3 : ([\{\}, \{\}] \vdash c_2),$ $l_1 : ([\{\}, \{a, e\}] \vdash p)$	$l_{2.1} \rightsquigarrow l_1,$ $l_3 \rightsquigarrow l_{2.1},$ $l_1 \rightsquigarrow \emptyset$

**Table 2.** Structured AB-dispute derivation for example 1.

The following realistic example illustrate the possible use of structured AB-dispute derivations in real-world setting. Imagine a scenarios whereby parents are trying to find a name for their new-born baby. Let us assume that parents deem a name as acceptable if they both like it and it is easy to remember. Also, Dad dislikes common names and he also does not want the baby to have the same name as his uncle. Mom on the other hand by default hates all names unless she explicitly approves of them. This scenario can be expressed as the following ABA framework<sup>3</sup>:

$\mathcal{L} = \{p(t) \mid p \in \text{Preds and } t \in \text{Terms}\}$  where *Preds* are all predicates occurring in  $\mathcal{R}$  and  $\text{Terms} = \{\text{adrian}, \text{vercingetorix}\}$

$$\mathcal{A} = \{\text{all\_like}(Name), \text{mom\_hates}(Name) \mid Name \in \text{Terms}\}$$

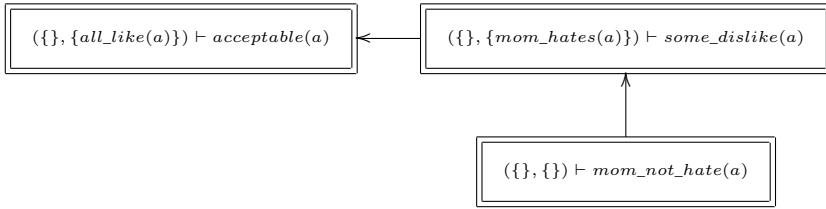
$$\mathcal{R} = \left\{ \begin{array}{l} \text{acceptable}(Name) \leftarrow \text{all\_like}(Name), \text{easy\_to\_remember}(Name); \\ \text{easy\_to\_remember}(Name) \leftarrow \text{short}(Name); \\ \text{some\_dislike}(Name) \leftarrow \text{mom\_hates}(Name); \\ \text{some\_dislike}(Name) \leftarrow \text{dad\_hates}(Name); \\ \text{dad\_hates}(Name) \leftarrow \text{too\_commom}(Name); \\ \text{dad\_hates}(Name) \leftarrow \text{uncle\_has}(Name); \\ \text{mom\_not\_hate}(Name) \leftarrow \text{mom\_said\_ok}(Name); \\ \text{mom\_said\_ok}(\text{adrian}); \\ \text{short}(\text{adrian}) \end{array} \right\}$$

<sup>3</sup>The rules containing variables, i.e. words beginning with an upper case letter, are short-hand for all their ground instances (e.g. all rules where *Name* is instantiated to *adrian*).

There exists a structured AB-dispute derivation of defence set  $D_{11}$  and of dialelectical structure  $(A_{11}, D_{11})$  for  $\text{acceptable}(\text{adrian})$  where

$$\begin{aligned} D_{11} &= \{\text{all\_like(adrian)}\} \text{ and } R_{11} = \{l_1 \rightsquigarrow \emptyset, l_2 \rightsquigarrow l_1, l_3 \rightsquigarrow l_2\} \text{ and} \\ A_{11} &= \{l_1 : ([\{\}], \{\text{all\_like(adrian)}\}) \vdash \text{acceptable(adrian)}), \\ &\quad l_2 : ([\{\}], \{\text{mom\_hates(adrian)}\}) \vdash \text{some\_dislike(adrian)}), \\ &\quad l_3 : ([\{\}], [\{\}] \vdash \text{mom\_not\_hate(adrian)})\} \end{aligned}$$

Note that all computed arguments are actual arguments. The dialectical structure can be graphically represented as follows (here  $\leftarrow$  stands for attack):



**Figure 1.** Dialectical structures for the realistic example.

#### 4. Results

In this section, we state and sketch formal proofs of the following main results<sup>4</sup>: 1) our structured AB-dispute derivations are a direct generalisation of the AB-dispute derivations of (a variant of) [7,8,9] (see theorem 1 below); 2) structured AB-dispute derivations compute admissible supports of the input claims (see corollary 1 below); 3) the dialectical structure computed by structured AB-dispute derivations can be mapped onto admissible dispute trees [7,9] (see theorem 2 below). We prove the results for ABA frameworks where the contrary of every assumption is a singleton set (as in original ABA). As discussed in section 2, this can be done without loss of generality.

**Theorem 1** *For any structured AB-dispute derivation of a defense set  $D$  and of dialelectical structure  $(A, R)$  for a claim  $\alpha$  there exists an AB-dispute derivation of a defense set  $D$  for  $\alpha$ .*

Note that the converse also holds, namely existence of an AB-dispute derivation guarantees existence of a structured AB-dispute derivation computing the same defense set (and some dialelectical structure).

The proof of theorem 1 uses a variant of AB-dispute derivation, equivalent to the one in [9]. This variant combines two cases in AB-dispute derivations in [9], cases 2ic.1 and 2ic.2, into one single case 2ic, corresponding to case 2ic in definition 3. This variant is equivalent to the original version under the restriction that, when the contrary of an assumption is an assumption, its original contrary is the original assumption. This restriction also held (implicitly) in [9]. The proof of theorem 1 is constructive, in that it uses mappings from the  $\mathcal{P}_i$  and  $\mathcal{O}_i$  components in structured AB-dispute derivations onto  $\mathcal{P}_i$  and  $\mathcal{O}_i$  components in AB-dispute derivations: these mappings turn sets of po-

<sup>4</sup>Full proofs can be found in an accompanying technical report.

tential arguments into sets of assumptions (for  $\mathcal{P}_i$ ; these are all the assumptions in the unmarked part of the support of the potential arguments) and sets of sets of assumptions (for  $\mathcal{O}_i$ ; each such set is the unmarked part of the support of one potential argument). No mappings are required for the  $D_i$  and  $C_i$  components - that are identical in the two kinds of derivations, or the  $A_i$  and  $R_i$  components, that are absent in the original AB-dispute derivations.

**Corollary 1** *Given any structured AB-dispute derivation of a defense set  $D$  and of a dialectical structure  $(A, R)$  for a sentence  $\alpha$ ,  $D$  is an admissible support for  $\alpha$ .*

This result follows directly from theorem 1 and the correctness of AB-dispute derivations in [7,9].

Theorem 2 below sanctions that the dialectical structure computed by structured AB-dispute derivations can be mapped onto admissible dispute trees [7,9], defined as follows.

**Definition 4** (*definitions 3.1 and 3.2 in [9]*) *A dispute tree for an argument  $a$  is a (possibly infinite) tree  $T$  such that*

1. *Every node of  $T$  is labelled by an argument and is assigned the status of proponent node or opponent node, but not both.*
2. *The root is a proponent node labelled by  $a$ .*
3. *For every proponent node  $N$  labelled by an argument  $b$ , and for every argument  $c$  that attacks  $b$ , there exists a child of  $N$ , which is an opponent node labelled by  $c$ .*
4. *For every opponent node  $N$  labelled by an argument  $b$ , there exists exactly one child of  $N$  which is a proponent node labelled by an argument which attacks  $b$ .*
5. *There are no other nodes in  $T$  except those given by 1-4 above.*

*The set of all assumptions belonging to the proponent nodes in  $T$  is called the defense set of  $T$ . A dispute tree is admissible iff no argument labels both a proponent and an opponent node.*

Given a dialectical structure  $(A, R)$  computed by a structured AB-dispute derivation, let  $Actual(A, R)$  stand for the pair  $(A^*, R^*)$  consisting of the set  $A^*$  of all actual arguments that can be obtained from  $A$  (by backward deduction from the potential arguments in  $A$ ) and  $R^*$  the restriction of  $R$  to elements of  $A^*$ . Moreover, given a dialectical structure  $(Args, Rel)$ ,  $T(Args, Rel)$  will refer to the tree constructed as follows: the root is the argument attacking  $\emptyset$  in  $Args$ ; nodes are in correspondence with elements of  $Args$ ;  $x$  is a child of  $y$  iff  $(x, y) \in Rel$ . Then,

**Theorem 2** *For any structured AB-dispute derivation of a defense set  $D$  and of dialectical structure  $(A, R)$  for a claim  $\alpha$ , let  $Actual(A, R) = (A^*, R^*)$  and  $T = T(A^*, R^*)$ . Then,  $T$  is an admissible dispute tree for  $\alpha$  with defense set  $D'$  such that  $D' \subseteq D$ .*

The proof of this theorem relies upon a number of lemmas, including:

**Lemma 1** *For each structured AB-dispute derivation of a defense set  $D$  and of dialectical structure  $(A, R)$  for a claim  $\alpha$ , let  $C$  be the final set of culprits. For every  $x \in C$  there exists an argument in  $A$  attacking  $x$ .*

**Lemma 2** *For each structured AB-dispute derivation of a defense set D and of dialectical structure (A, R) for a claim  $\alpha$ , all the actual arguments that one can build from any potential argument in A are attacked by some argument in A.*

**Lemma 3** *For each structured AB-dispute derivation of a defense set D and of dialectical structure (A, R) for a claim  $\alpha$ , let C be the final set of culprits. Every potential argument in A of the form  $(S_u, S_m \vdash x)$  such that  $S_u \neq \{\}$  has the property that  $(X_u \cup X_m) \cap C \neq \{\}$ .*

## 5. Conclusions

We have presented a computational model for a form of argumentation that is a hybrid between abstract and assumption-based argumentation. To the best of our knowledge, this work is the first attempt to combine the two forms of argumentation in a synergetic and practical manner, building upon [7,9,11]. Our hybrid computational model would be beneficial to developing applications of argumentation where explicit justifications of claims in terms of full dialectical structures are required, for example, for the purpose of argumentation-based negotiation, as it would provide the negotiating agents with an explicit structure of the dialectical process.

Computational models have been proposed for abstract argumentation, such as the Two Party Immediate response (TPI) disputes [14] and the dialectical proof theories of [4,13]. Like ours, these models can be seen as games between two fictional players in which the proponent always acts first. These models compute different semantics than admissibility (i.e. preferred [4,14], robust and defensible [13] semantics) for a different argumentation framework (i.e. abstract argumentation). In particular, these computational models do not need to construct arguments, as arguments are seen as black boxes within abstract argumentation. Moreover, although they use a dialectical protocol similar to the one underlying the generalised AB-dispute derivation, [4,13] insist on and [14] implies strictly alternating turns between proponent and opponent, whereas we do not do so in order to allow interleaving the construction of arguments and the analysis of their dialectical status.

Compared to existing computational models for assumption-based argumentation (e.g. [9]), our computational model manipulates potential, rather than actual, arguments. These correspond to stages in the construction of actual arguments, and allow the interleaving of the construction of (actual) arguments and their evaluation (with respect to the admissibility semantics). As a result, the set of arguments returned by our computational model may include potential arguments, that have been defeated before being fully constructed. This may seem as a departure from conventional abstract argumentation. Note however that the computational model relies upon a selection function for deciding how arguments are constructed, using backward deductions. As noted in [7], when this selection function is *patient*, the computed arguments are all guaranteed to be actual. In other words, our computational model generalises conventional abstract argumentation without changing its spirit.

Structured AB-dispute derivations have been implemented in the CaSAPI (Credulous and Sceptical Argumentation: Prolog Implementation) system (in its current version v4.3) which can be downloaded from [www.doc.ic.ac.uk/~dg00/casapi.html](http://www.doc.ic.ac.uk/~dg00/casapi.html). Previous

versions were inspired by traditional assumption-based argumentation (version 2, [10]) and by a restricted form of hybrid argumentation (version 3, [11]).

In the near future we aim at improving the practical aspects of hybrid argumentation, by extensive experimentation with the CaSAPI system and by extending its graphical user interface. We also plan to compare this system with a number of other argumentation systems, including Gorgias [5], for credulous argumentation in argumentation frameworks with preferences amongst defeasible rules, the ASPIC system [3] dealing with quantitative uncertainty, DeLP [12] for defeasible logic programming, and the system by Krause et al. [2]. A mapping from these various frameworks onto assumption-based argumentation (possibly extended) is needed in order to carry out a full comparison.

Finally, we also plan to extend (theoretically and practically) the computational machinery in this paper to compute sceptical semantics for argumentation, namely the grounded and ideal semantics, already implemented in version 2 of CaSAPI.

## Acknowledgements

This work was partially funded by the Sixth Framework IST programme of the EC, under the 035200 ARGUGRID project.

## References

- [1] A. Bondarenko, P.M. Dung, R.A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1-2):63–101, 1997.
- [2] D. Bryant and P. Krause. An implementation of a lightweight argumentation engine for agent applications. In *Proc. of JELIA06*, 2006.
- [3] M. Caminada, S. Doutre, S. Modgil, H. Prakken, and G.A.W. Vreeswijk. Implementations of argument-based inference. In *Rev. of Argumentation Tech.*, 2004.
- [4] Claudette Cayrol, Sylvie Doutre, and Jérôme Mengin. On Decision Problems Related to the Preferred Semantics for Argumentation Frameworks. *Journal of Logic and Computation*, 13(3):377–403, 2003.
- [5] N. Demetriou and A. C. Kakas. Argumentation with abduction. In *Proceedings of the fourth Panhellenic Symposium on Logic*, 2003.
- [6] P.M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
- [7] P.M. Dung, R.A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Art. Int.*, 170:114–159, 2006.
- [8] P.M. Dung, P. Mancarella, and F. Toni. A dialectic procedure for sceptical, assumption-based argumentation. In *Proc. of the 1st Int'l Conference on Computational Models of Argument*, 2006.
- [9] P.M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence - Special Issue on Argumentation in Artificial Intelligence*, 171(10-15):642–674, July–October 2007.
- [10] D. Gaertner and F. Toni. CaSAPI: a system for credulous and sceptical argumentation. In *Proc. LPNMR Workshop on Argumentation for Non-monotonic Reasoning*, pages 80–95, 2007.
- [11] D. Gaertner and F. Toni. Computing arguments and attacks in assumption-based argumentation. *IEEE Intelligent Systems*, 22(6), 2007.
- [12] A. Garcia and G. Simari. Defeasible logic programming: An argumentative approach. *Journal of Theory and Practice of Logic Prog.*, 4(1-2):95–138, 2004.
- [13] Hadassa Jakobovits and Dirk Vermeir. Dialectic semantics for argumentation frameworks. In *International Conference on Artificial Intelligence and Law*, pages 53–62, 1999.
- [14] G. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proc. JELIA*, volume 1919 of *LNCS*, pages 224–238. Springer Verlag, 2000.

# Requirements for reflective argument visualization tools: a case for using validity as a normative standard

Michael H.G. HOFFMANN<sup>1</sup>  
*Georgia Institute of Technology, USA*

**Abstract.** This paper formulates in the first part some requirements for a certain sort of computational argumentation systems, namely those which are designed for a very specific purpose: to motivate reflection on one's own thinking, and to induce cognitive change. This function of argumentation systems is important for argument-based conflict negotiations, deliberation processes, intercultural communication, text analysis, and learning through argument visualization. In all these situations success is only possible when people are able to change their mind, learn something, or start to reframe well-established ways of perceiving and interpreting things. Based on these requirements, I defend and explain in the second part my decision to use for Logical Argument Mapping—a method specifically designed for supporting reflective argumentation—only argument schemes that are deductively valid.

**Keywords.** Semiautomated computational argumentation systems, Computer Supported Argument Visualization (CSAV), Logical Argument Mapping (LAM), cognition, diagrammatic reasoning, Peirce

## Introduction

Visualizing arguments is not only useful when we want to persuade somebody, but also when we want to clarify our own thinking about “wicked” problems [1], or when we use arguments in social settings to facilitate, for example, negotiations in conflicts [2], deliberation processes, or intercultural communication. In all these situations a main function of argument visualization is to stimulate reflection and cognitive change; the goal is to *learn* something. If arguments fulfill primarily this function, I use the term “reflective argumentation.” Since I think that reflective argumentation is something that everybody should learn, I am interested in computational argument systems that could support reflective argumentation not only for specialists, but for a broader community of users including, let's say, high school students.

This paper is mainly about the requirements for those argumentation systems that are supposed to stimulate reflection. In its second part, however, I will argue that these requirements could best be fulfilled by systems of argumentation that focus particularly on deductively valid argument schemes.

---

<sup>1</sup> School of Public Policy, Georgia Institute of Technology, 685 Cherry Street, N.W., Atlanta, GA 30332-0345, USA; E-mail: m.hoffmann@gatech.edu.

## 1. Requirements

Focusing on what I call “reflective argumentation” means, first of all, that an argumentation system has to fulfill the following **first basic requirement**: It must allow the visualization of arguments in a way that what the user sees when looking at the construction of her or his argument compels her or him to reflect critically on the conditions and presumptions that guided the process of argument construction itself. Cognitive change must be induced, as Kathleen Hull puts it in her description of Charles’ Peirce’s concept of “diagrammatic reasoning,” by something that—although created by ourselves—“stands up against our consciousness” ([3], 282, 287; cf. [4], CP 1.324; [5]).

This basic requirement leads immediately to a criticism of two extremes that can be distinguished in the broad spectrum of computer-based argument support systems that are available today. On the one hand, there are systems that do not challenge the user enough to formulate *complete* arguments (e.g. [6], [1], [7]); the problem is here: if premises remain hidden, there is no way to reflect on them. On the other hand, there are systems that—while trying to represent everything—are so complex and abstract that the user is more threatened by confusion than by reflection (e.g. [8], [9], [10]). Obviously, there is a trade-off between user-friendliness on one hand, and precision, disambiguation, systematicness, completeness, and range of applications on the other [11].

Talking about user-friendliness and this trade-off means first of all that we need to *identify* the specific problems normal people have with arguments and argumentations. (I am using “argument” here as a set of statements—a claim and one or more reasons—where the reasons jointly provide support for the claim, or are at least intended to support the claim. An “argumentation” is defined here as a set of arguments in which a main argument is supported by further arguments). Empirical research on Computer Supported Collaborative Learning (CSCL) and Computer Mediated Communication (CMC) indicates that computer tools often produce “technical disturbances and a loss of thematic focus” that “have a negative effect on collaborative learning processes” ([12], 69). Sometimes, they cause people to talk more about the technique than about the task in front of them, increasing thus the cognitive load they have to handle instead of reducing it ([11], [13]). But even if no computer is involved, there is some anecdotic evidence from classroom experiences in the area of planning and public policy that hints at a deeper problem. When it comes to arguments, this area of research and teaching seems to be completely in the hands of the famous “Toulmin model” (e.g. [14]). However, as Gasper and George showed in a critical reflection on “assessing, improving, and transcending the Toulmin model” that was based on an analysis of published literature and classroom experiences in planning and public policy, there is a real danger of “oversimplification” since students often try “to squeeze everything into a single simple diagram” [15]. But if this is possible—and my own experience with teaching similar approaches to argumentation confirms this observation—then the deeper problem seems to be that many students simply do not know what an argument *is*, or how to apply an only vague notion of “argument” in concrete situations.

If this is indeed an adequate characterization of a quite substantial deficit in some parts of the world regarding a competence that should be an essential prerequisite for being a politically mature citizen at least in democratic societies, then we should determine a **second basic requirement** for reflective argument systems as follows: These systems should be designed in a way that the *education* of their users is part of the system itself. Simon Buckingham Shum is quite right when he says with regard to argument visualization tools that we need “a new *literacy* in being able to read and write in

the new medium, and a new *fluency* in using these conversational tools in appropriate ways in different contexts” ([16], 19). But since this will be true for prospective teachers of the new tools as well, it should be important to make the tools themselves educational.

The need of making the education of its users a part of an argument system implies a series of further requirements. The most important one is the following, **third requirement**: The system itself must provide the means users need to *evaluate* what they are doing when constructing an argument. This again leads to a **fourth requirement**: An argumentation system should provide a limited list of clearly defined argument schemes that function as the normative standard in reference to which a user can evaluate the completeness and strength of an argument. This is absolutely crucial for those systems of argumentation whose main purpose is to motivate reflection and to induce cognitive change. There are, of course, several ways to establish such a normative standard, but without it a user could never experience the “compulsory perceptions” Peirce was talking about when he emphasized that learning by experimenting with diagrams is only possible when these diagrams are constructed by means of a “representational system”—that is by a system with a clearly defined ontology and rules of operation—that determines the necessary outcome of such an experimentation. Peirce showed—mainly with regard to proofs in mathematics—that the normativity of those representational systems is the essential precondition of what I discussed in my first basic requirement as the compelling force of external representations: “if one exerts certain kinds of volition [in constructing a diagram, M.H.], one will undergo in return certain compulsory perceptions. .... certain lines of conduct will entail certain kinds of inevitable experiences” ([4], CP 5.9; see [5], [17] for some examples). For any learning by means of visualizations it is essential, I would argue, that what we see when looking at such a visualizations *forces* us to reconsider, to change, or to differentiate what we already know. Otherwise there would simply be no learning. In order to generate such an external force, however, it is essential—and this should be my **fifth requirement**—that we as the users of an argumentation system have to *accept* the normative character of its rules as something that is beyond our own power. And since an external representation is only compelling when we understand and realize the rules of the representation system in which it is constructed, people need to *learn*, and to train, how to meet a pre-defined normative standard of argumentation as strictly as possible.

A **sixth requirement** for argumentation systems that are supposed to support reflective argumentation has been mentioned already in connection with my first requirement under the heading of “completeness”: Whatever is relevant for the possibility of cognitive change, or what might have an impact on the acceptability of an argument, must be *visible* in the representation of the argumentation. This point is important since most of the arguments used in everyday contexts are enthymemes, that is incomplete arguments in which either one of the premises or even the conclusion is only implicitly given. The crucial question at this point is what exactly the meaning of “complete” is. Based on my personal observations of students trying to construct, to identify, or to reconstruct an argument without much preparation, I am inclined to say that the main problem of non-experts in argumentation is their readiness to accept nearly everything as a reason for something else as long as there is at least some relation between the two. What is missing is first of all an understanding that—as has been emphasized by Toulmin—a reason is only a reason for a claim if there is a “warrant” that “authorizes” the step from the reason to the claim ([18], 91). The distinction between “reason” and “warrant” is crucial when it comes to the evaluation of an argument since a critique can

always attack two different things that can be defended independently of each other: the truth of the *reason*, and the truth of the *warrant*. For this reason, a “complete” visualization of an argument must always include both, the formulation of a reason *and* the formulation of a corresponding warrant; if one of them is not already explicitly given, it has to be reconstructed.

At this point in my list of requirements for argumentation systems that should be able to support reflection and cognitive change, I think the best way to fulfill all the six requirements—and the easiest for users—is to use only *logically valid* argument schemes for the core arguments of a more complex argumentation, as realized in Logical Argument Mapping (LAM), an argument visualization method that I described elsewhere.<sup>2</sup> This decision can be defended with reference to each of the requirements discussed so far:

1. Since logical validity represents a clear normative standard, a system that allows only valid arguments would indeed “compel” the user “to reflect critically on the conditions and presumptions” that guide the process of argument construction
2. Such a system could easily “educate” the user to do it right
3. It would provide all the means a user needs to “evaluate” arguments
4. Logical valid argument schemes can be presented in a limited and clearly defined list
5. Since the validity of an argument scheme is either obvious or easily to demonstrate, it is easy for users to “accept” validity as normative standard
6. Since a valid argument is always a “complete” argument, accepting validity as normative standard helps the user to achieve completeness in constructing and evaluating argument.

## 2. Validity and defeasibility

Given the direction in which argumentation theory developed since Toulmin, the decision to use only argument schemes that are deductively valid is obviously controversial, to say the least. However, in addition to the arguments for this decision listed above, I want to emphasize one further point. Logical Argument Mapping builds on a clear separation between the *construction* of arguments a user performs in interaction with a software-based argumentation system and the *reflection* he or she performs for him- or herself either individually or in groups with regard to the acceptability of reasons and warrants. While the argument construction is determined and becomes evaluated by the criterion of logical validity, the reflection is determined by the *epistemological* question whether the reasons and warrants are true, or at least acceptable in the respective social situation.

This distinction between construction and reflection corresponds roughly to the differentiation between a “logical” and a “dialectical layer” that has been proposed by Henry Prakken and Giovanni Sartor for the description of legal arguments [20]. While the “logical layer defines what arguments are, i.e., how pieces of information can be combined to provide basic support for a claim,” the “dialectical layer focuses on conflicting arguments: it introduces such notions as ‘counterargument’, ‘attack’, ‘rebuttal’

---

<sup>2</sup> [2], [17], [19]. The most recent description is available at <http://www.prism.gatech.edu/~mh327/LAM>.

and ‘defeat’, and it defines, given a set of arguments and evaluation criteria, which arguments prevail” (344). Prakken and Sartor add to this a “procedural layer” and a “strategic or heuristic one,” but for my purposes it would be sufficient to subsume both of these under an enlarged understanding of “dialectic.” With regard to what I discuss here under the heading of reflective argumentation, the main idea is a clear separation between the *form* of arguments whose completeness can best be evaluated when they are constructed as logically valid arguments, and a *critical reflection and debate* on the truth of the premises of those arguments, either individually or in collaboration with others. This reflection would then lead either to a revision or refinement of the original argument, or to a counterargument, but again an argument that has to meet the normative standard defined by the chosen argumentation system. This way, the process of argumentation is conceived of as an *iterative* process of construction and reflection, a process that challenges the user to engage in a kind of dialectical process that leads her back and forth between improving her own understanding of the issue in question and the way she represents it.

Based on this procedural dialectic between argument construction and reflection, I would say that Logical Argument Mapping belongs to what has been discussed in the literature as “defeasible argumentation,” as long as defeasible argumentation is defined—as proposed by Henry Prakken and Gerard Vreeswijk in their handbook article on the “Logics of defeasible argumentation”—as a *process* in which “arguments for and against a certain claim are produced and evaluated, to test the tenability of a claim” ([21], 219). Since the goal of any sort of reflective argumentation is learning and the development of positions, reflective argumentation is defeasible *per definitionem*.

In general, however, the term “defeasible” is first of all used in argumentation theory as the *essential property* of a certain class of arguments. Douglas Walton, for example, distinguishes in his textbook *Fundamentals of Critical Argumentation* “deductive,” “inductive,” and “defeasible inferences,” defining the last group as those that “may turn out to fail (default) if new evidence comes in” ([22], 52). Or, as Prakken and Sartor write: A defeasible argument is “an argument that is acceptable in itself,” but “can be overturned by counterarguments” ([20], 342). The classical example is the argument: “Birds fly; Tweety is a bird; therefore, Tweety flies” [23]. This argument can obviously be defeated when we think about penguins, dead birds, birds whose feet are encased in concrete, and hundreds of other possible exceptions that would refute a universal statement like “all birds can fly.”

The crucial point, however, is the following. Walton writes at one point that a defeasible inference “is *inherently* subject to retraction” ([22], 52; my italics). If defeasibility is an “inherent” property of a certain class of arguments, then it must be possible to decide in each case whether this property is given or not. But how could that be possible? What exactly is the criterion that allows an objective and clear-cut distinction between defeasible and non-defeasible arguments? Although it might be intuitively convincing that “all birds can fly” is defeasible while “all humans are mortal” is not, it is obviously an *epistemological* question where to draw the line; it is the question what we are willing to accept as true based on the knowledge that is available in a certain situation. But the point is: As we know from the history of science, what we “know” is in permanent change. Yesterday, every specialist believed “that if people with diabetes lowered their blood sugar to normal levels, they would not longer be at high risk of dying from heart disease,” and today we learn that a major study has found that lowering blood sugar with people with type 2 diabetes “actually increased their risk of death” [24]. As has been shown in 20<sup>th</sup> century’s philosophy of science time and again, it is

simply impossible to justify *any* knowledge claim in a way that future revisions or refutations are excluded.

My impression is that the whole discussion about defeasible arguments confuses one of the most important distinctions in logic, namely the distinction between validity and soundness. As everyone knows, for validity the truth of the premises of an argument is simply *presupposed*; truth is only important when it comes to the soundness of an argument. That means, however, that *any* deductively valid argument whatsoever can be conceived of as a defeasible argument, because defeasibility concerns only the truth of the premises. While validity is a question of logic, truth is a question of epistemology and science. As long as we accept both a clear distinction between validity and soundness and an understanding of argumentation that builds on the dialectic between argument construction and reflection, there is no problem to establish logical validity as a normative standard of argument construction, while at the same time admitting that any premise of a logically valid argument can be criticized, refined, defeated, and abandoned at any time.

A deeper reason, I guess, for the controversy between approaches to argumentation that include defeasible inferences and those that are more restrictive concerns different ideas about which *purposes* models of argumentation are supposed to fulfill. In their handbook article, Prakken and Vreeswijk write with the regard to the more general research on nonmonotonic reasoning—in which they include systems for defeasible argumentation—that most “nonmonotonic logics aim to formalize” the “phenomenon of ‘default reasoning’,” that is reasoning that is based on “a conditional that can be qualified with phrases like ‘typically’, ‘normally’ or ‘unless shown otherwise’” [21]. Formalizing default reasoning is of course important when the goal is to simulate, or to model, human reasoning. But this is not the goal I have in mind. I am only interested in *tools* people can use to improve their thinking, that is tools for reflective argumentation. A tool like our language. There is no need that the *grammar* of a language must be very complex to provide a powerful tool. And there is no need to provide more than a bunch of deductively valid argument schemes when the main objective is to engage users in an iterative process of argument construction, reflection, reconstruction, and so on. There is simply no need to define all the possible exceptions to a universal statement like “all birds can fly” in advance, or certain general questions that always come up with certain argument schemes ([25], [8]), when we can expect users who can contribute for themselves—based on a critical reflection on the elements of an argument—those considerations that would be necessary to improve it. It is, of course, an open question which argument schemes we need to guarantee a good balance between user-friendliness and the sophistication of an argumentation system, but the general maxim should be: keep the grammar of your argument language as simple as possible, and as rich as necessary.

In Logical Argument Mapping, for example, arguments from induction, analogy, and expert opinion would be *constructed* in the logically valid forms of “complete induction,” “perfect analogy,” and “perfect authority.”<sup>3</sup> Although it is obviously hard to find any of these forms in real life situations, this does not matter much as long as the *user* becomes challenged to reflect on the problems that are inevitably involved with these argument schemes. It should be better to confront the user directly with the *problems* of these argument schemes and to challenge further reflections than risking that implicit assumptions remain unrevealed.

<sup>3</sup> See <http://www.prism.gatech.edu/~mh327/LAM>.

This way, it is even possible to represent abductive inferences by means of LAM. Abduction has first been defined by Peirce as “the process of forming an explanatory hypothesis. It is the only logical operation which introduces any new idea” ([4], CP 5.171). “Logical,” of course, is used by Peirce in a much broader sense than deductively valid [26]; abduction, he says at one point, “consists in examining a mass of facts and in allowing these facts to suggest a theory. In this way we gain new ideas; but there is no force in the reasoning” (CP 8.209). Peirce describes the inferential form of abduction as follows: “The surprising fact, C, is observed; But if A were true, C would be a matter of course, Hence, there is reason to suspect that A is true” (CP. 5.189). Although the conclusion of an abductive inference is only a hypothesis, it is clear that *if* this hypothesis would be true, as well as the assumption that the observed fact is implied by this hypothesis, the whole argument could be translated into a simple *modus ponens* argument. This means, however, that a user can at any time represent an abductive assumption as a *modus ponens* in order to reflect *then* on the question whether both the premises of this argument can be accepted as true.

The crucial point that distinguishes my approach to argumentation from the mainstream is that I am not interested to “model” everyday, commonsense reasoning. Of course, it might be an interesting task to develop software that “mirrors” somehow the way people argue for or against a position in different contexts. But why should it not also be an interesting task to develop argumentation systems that can be used simply as *tools* for normal people? Nobody expects that a calculator “models” what humans are doing when they compare prices in the supermarket. A tool is always something that *augments* our natural abilities. Nobody would measure the quality of a tool by its similarity to what we can do anyway, but by its functionality and effectiveness for a predefined purpose. Since the objective of my work is to develop tools that stimulate reflection and learning, the quality of these tools should not be assessed by their degree of similarity to human processes of reflection and learning, but by its success in fulfilling this purpose.

## References

- [1] Kirschner PA, Buckingham Shum SJ, Carr CS, editors. Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making. London: Springer; 2003.
- [2] Hoffmann MHG. Logical argument mapping: A method for overcoming cognitive problems of conflict management. International Journal of Conflict Management 2005;16(4):304-334.
- [3] Hull K. Why Hanker After Logic? Mathematical Imagination, Creativity and Perception in Peirce's Systematic Philosophy. Transactions of the Charles S. Peirce Society 1994;30:271–295.
- [4] Peirce. Collected Papers of Charles Sanders Peirce. Cambridge, Mass.: Harvard UP; CP.
- [5] Hoffmann MHG. How to Get It. Diagrammatic Reasoning as a Tool of Knowledge Development and its Pragmatic Dimension. Foundations of Science 2004;9(3):285-305.
- [6] Conklin J. Dialogue Mapping: Building Shared Understanding of Wicked Problems. Chichester, England; Hoboken, NJ: John Wiley & Sons; 2006.
- [7] Okada A, Buckingham Shum S, Sherborne T, editors. Knowledge Cartography. London: Springer; 2008.
- [8] Gordon TF, Prakken H, Walton D. The Carneades model of argument and burden of proof. Artificial Intelligence 2007;171(10-15):875-896.
- [9] Prakken H. Formal systems for persuasion dialogue. Knowledge Engineering Review 2006;21(2):163-188.
- [10] Rahwan I, Zablith F, Reed C. Laying the foundations for a World Wide Argument Web. Artificial Intelligence 2007;171(10-15):897-921.
- [11] van Bruggen JM, Boshuizen HPA, Kirschner PA. A Cognitive Framework for Cooperative Problem Solving with Argument Visualization. In: Kirschner PA, Buckingham Shum SJ, Carr CS, editors. Visu-

- alizing Argumentation: Software Tools for Collaborative and Educational Sense-making. London: Springer; 2003. p 25-47.
- [12] Kanselaar G, Erkens G, Andriessen J, Prangsma M, Veerman A, Jaspers J. Designing Argumentation Tools for Collaborative Learning. In: Kirschner PA, Buckingham Shum SJ, Carr CS, editors. Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making. London: Springer; 2003. p 51-70.
- [13] van Bruggen JM, Kirschner PA, Jochems W. External representation of argumentation in CSCL and the management of cognitive load. *Learning and Instruction* 2002;12(1):121-138.
- [14] Dunn WN. Public Policy Analysis: An Introduction. Englewood Cliffs, NJ: Prentice Hall; 2007 <1981>.
- [15] Gasper DR, George RV. Analyzing argumentation in planning and public policy: assessing, improving, and transcending the Toulmin model. *Environment and Planning B-Planning & Design* 1998;25(3):367-390.
- [16] Buckingham Shum S. The Roots of Computer-Supported Argument Visualization. In: Kirschner PA, Buckingham Shum SJ, Carr CS, editors. Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making. London: Springer; 2003. p 3-24.
- [17] Hoffmann MHG. Logical Argument Mapping: A cognitive-change-based method for building common ground. ACM International Conference Proceeding Series; Vol. 280. Proceedings of the 2nd international conference on Pragmatic web 2007:41-47.
- [18] Toulmin S. The uses of argument. Cambridge, U.K.; New York: Cambridge UP; 2003 <1958>.
- [19] Hoffmann MHG. Analyzing Framing Processes By Means Of Logical Argument Mapping. In: Donohue B, Kaufman S, Rogan R, editors. *Framing in Negotiation: State of the Art*. 2008. Forthcoming.
- [20] Prakken H, Sartor G. The role of logic in computational models of legal argument: A critical survey. *Computational Logic: Logic Programming and Beyond, Pt II*. Volume 2408, Lecture Notes in Artificial Intelligence. 2002. p 342-381.
- [21] Prakken H, Vreeswijk G. Logics of defeasible argumentation. In: Gabbay DM, Guenther F, editors. *Handbook of philosophical logic*. 2nd ed. Volume IV. Dordrecht; Boston: Kluwer Academic Publishers; 2001. p 219-318.
- [22] Walton D. Fundamentals of Critical Argumentation. Cambridge; New York: Cambridge UP; 2006.
- [23] Reiter R. Nonmonotonic reasoning. *Annual Review of Computer Science* 1987;2:147-186.
- [24] Kolata G. Diabetes Study Partially Halted after Deaths. *The New York Times*, 2008, February 7.
- [25] Walton D. Appeal to Expert Opinion. University Park: Penn State Press; 1997.
- [26] Hoffmann MHG. Problems with Peirce's Concept of Abduction. *Foundations of Science* 1999;4(3):271-305.

# Argumentation Using Temporal Knowledge

Nicholas MANN<sup>a</sup>, Anthony HUNTER<sup>a</sup>

<sup>a</sup>Department of Computer Science, University College London, Gower Street, London, WC1E 6BT, UK.

**Abstract.** Proposals for logic-based argumentation have the potential to be adapted for handling diverse kinds of knowledge. In this paper, a calculus for representing temporal knowledge is proposed, and defined in terms of propositional logic. This calculus is then considered with respect to argumentation, where an argument is pair  $\langle \Phi, \alpha \rangle$  such that  $\Phi$  a minimally consistent subset of a database entailing  $\alpha$ . Two alternative definitions of an argument are considered and contrasted.

**Keywords.** Argumentation systems, Arguments, Temporal argumentation.

## 1. Introduction

Argumentation is a powerful tool for reasoning with inconsistent knowledge in a database. There are many examples and types of argumentation (see reviews [1,2,3]). These argumentation systems range in style from the abstract argumentation system of Dung [4], through defeasible argumentation of Garcia and Simari's DeLP [5] to the classical logic approach of Besnard and Hunter's argumentation system [6].

A common definition of an argument is a that of a minimally consistent subset of a database that is capable of proving a given conclusion (eg [6,7]).

**Definition 1.1.** An **argument** is a pair  $\langle \Phi, \alpha \rangle$ , such that:

- (1)  $\Phi \not\models \perp$
- (2)  $\Phi \vdash \alpha$
- (3)  $\Phi$  is a minimal subset of a database  $\Delta$  satisfying (2)

We say  $\langle \Phi, \alpha \rangle$  is an argument for  $\alpha$ , with  $\Phi$  being the support of the argument, and  $\alpha$  being the conclusion of the argument.

In this definition, classical propositional logic is usually used. However, to express temporal knowledge, we need to extend this definition, and replace propositional logic. One simple idea here is to use first order logic (eg [8]), but full first order logic is a large step to take in order to express temporal knowledge where a simpler definition may suffice. Hence, in this paper we consider some of the space between the established use of propositional logic and the use of first order logic within the above definition.

With the replacement of propositional logic in definition 1.1,  $\vdash$  and  $\perp$  may have different meanings. This may provide problems, since for example established theorems

may no longer hold, or perhaps an inconsistent subset of the knowledgebase may still prove to be useful as a support. However, these changes also present opportunities to expand upon the definition of an argument and broaden our horizons as to what an argument could be.

To be more specific, this paper uses a calculus capable of expressing temporal knowledge. However, this in turns brings more problems; types of temporal knowledge are diverse, as are the methods of formalising them, and many possible logics and calculuses are available, such as the event calculus of Kowalski and Sergot [9] or the tense logic of Prior [10].

Before picking a method of expressing temporal knowledge, it is useful to consider the type of knowledge we wish to express, and for this purpose we use the example of business news reports, which for example could include phrases such, “Cisco Systems announced year on year sales growth of 12% for Q1”. News reports are interesting for several reasons; they can have conflicting information (usually in the form of several different sources providing different information), making them suitable for consideration in an argumentation system. They also have a practical application in terms of providing knowledge for which decisions need to be made; for example, the above sentence would provide useful information regarding a decision as to whether to buy Cisco stock. Most importantly with respect to this paper, though, is that they draw heavily on temporal knowledge due to their day-to-day nature - what is true today may well not be true tomorrow - as well as their inclusion of information concerning periods of time, such as profitability in the current or previous quarter.

Given the use of news reports, we can identify several possible features that we would wish to express in our database; most specifically that news reports concerning business are frequently about properties that change over an interval in time, such as recent sales growth or employment figures. We also require some measure of simplicity if we are to use this with argumentation; since little has been done in the way of temporal argumentation, we should choose a logic as close to the familiar as possible, and with as few new features as possible. However, we also require a certain amount of expressive power. Given these requirements, in this paper we propose a calculus built upon the ideas of Allen’s interval logic [11]. Rather than using abstract intervals, and in keeping with the desire for a practical system, we restrict the system to using specific timepoints.

Before we go into the definition of our calculus, it should be noted that there are two other approaches to temporal argumentation can be found in the literature; that of Hunter [12] and Augusto and Simari [13]. Both of these have a differing standpoint to that presented here. Hunter’s system is based on maximally consistent subsets of the knowledgebase, which are now not normally regarded as representative of arguments. Augusto and Simari’s contribution is based upon a many sorted logic with defeasible formulae, and hence also falls into a different category of argumentation, and the use of many sorted logic raises similar concerns to that of using first order logic.

## 2. The calculus $\mathcal{T}$

In this section, we present the syntax of a calculus  $\mathcal{T}$ , and the machinery needed to define the consequence relation  $\vdash_{\mathcal{T}}$ .

## 2.1. Syntax

Since we are working with fixed timepoints, we can define a timeline as below.

**Definition 2.1.** A **timeline** is a tuple  $\langle T, \preceq \rangle$ , where  $T$  is a finite set of **timepoints**, and  $\preceq$  is a total linear ordering over these timepoints. The symbol  $\prec$  is used such that  $t_1 \prec t_2$  iff  $t_1 \preceq t_2 \wedge t_1 \neq t_2$ .

**Definition 2.2.** An **interval** (on a timeline  $\langle T, \preceq \rangle$ ) is a pair  $(t_1, t_2)$  where  $t_1$  and  $t_2$  are both timepoints in  $T$ , and  $t_1 \prec t_2$ .

Since a finite subset of  $\mathbb{Z}$  and  $\leq$  are perfectly adequate, in this paper we shall use this definition. Using this we can define relationships between fixed intervals (as noted by Hamblin [14], with the remaining six relationships being the inverses of the first six).

**Definition 2.3.** Relationships between intervals are defined as follows:

- ( $a, b$ ) **precedes** ( $c, d$ ) iff  $b < c$
- ( $a, b$ ) **meets** ( $c, d$ ) iff  $b = c$
- ( $a, b$ ) **overlaps** ( $c, d$ ) iff  $a < c < b < d$
- ( $a, b$ ) **during** ( $c, d$ ) iff  $c < a < b < d$
- ( $a, b$ ) **starts** ( $c, d$ ) iff  $a = c$  and  $b < d$
- ( $a, b$ ) **ends** ( $c, d$ ) iff  $c < a$  and  $b = d$
- ( $a, b$ ) **equals** ( $c, d$ ) iff  $a = c$  and  $b = d$

For the letters of our calculus  $\mathcal{T}$ , we have a set of properties, given below as  $\alpha, \beta$  etc, but from a practical standpoint would be more like *sales*, *profits*, to represent the sales or profits for the company in question rising. For each of these, we state whether that property holds over an interval using *Holds*( $\alpha, i$ ) for some interval  $i$ , which may be a variable or fixed interval; for example, if we want to say that sales have risen during month 3, we could use *Holds*(*sales*, (2, 3)).

**Definition 2.4.** The syntax for the letters of  $\mathcal{T}$  is defined as below.

$$\begin{aligned}
symbol &::= \alpha \mid \beta \mid \gamma \mid \delta \dots \\
timepoint &::= 0 \mid 1 \mid 2 \mid 3 \dots \\
varinterval &::= i \mid j \mid k \dots \\
interval &::= (timepoint, timepoint) \\
&\quad \mid varinterval \\
relation &::= \text{precedes} \mid \text{meets} \mid \text{overlaps} \mid \text{during} \\
&\quad \mid \text{starts} \mid \text{ends} \mid \text>equals \\
letter &::= \text{Holds}(symbol, interval) \\
&\quad \mid (interval relation interval)
\end{aligned}$$

We can combine these with the usual propositional connectives ( $\wedge, \vee, \neg$ , and also for convenience  $\rightarrow$  and  $\leftrightarrow$ ). This is not strictly a classical propositional language however, due to the presence of variable intervals, and in the remainder of this chapter, we provide a method of translation between this calculus and classical propositional logic.

**Example 2.1.** The following are formulae in  $\mathcal{T}$ :

$$\begin{aligned}
 & \text{Holds}(\alpha, i) \\
 & ((\text{Holds}(\alpha, i) \wedge (i \text{ meets } j)) \vee \neg \text{Holds}(\beta, j)) \\
 & (((0, 1) \text{ starts } (4, 7)) \wedge \text{Holds}(\alpha, (3, 4)))
 \end{aligned}$$

Note the third formula is acceptable according to the syntax, although it appears (and will prove to be) a contradiction.

Within  $\mathcal{T}$  formulae, all variable intervals are universally quantified over the entire formula, hence a variable interval  $i$  refers to the same interval throughout a formula, although may be a different interval in a different formula.

## 2.2. Grounding

In order to work towards the conversion of formulae containing variable intervals into formulae in propositional logic, we consider the idea of a grounding.

**Definition 2.5.** A  $\mathcal{T}$  **grounding** (also just a grounding) is pair  $[\alpha, \Theta]$ , where  $\alpha$  is a formula in  $\mathcal{T}$  and  $\Theta$  is a set (possibly empty) of substitutions (the **substitution set**), in the form  $i/(a, b)$ , where  $i$  is a variable interval in  $\alpha$ ,  $(a, b)$  is any valid fixed interval (hence  $a < b$ ), and there is no other substitution for  $i$  in  $\Theta$ . If  $\Theta$  contains a substitution for all variable intervals in  $\alpha$ , then the pair is termed a **complete grounding**, otherwise it is a **partial grounding**.

**Example 2.2.** Valid groundings include

$$\begin{aligned}
 & [\text{Holds}(\alpha, i), \{i/(2, 3)\}] \\
 & [\text{Holds}(\alpha, i), \emptyset] \\
 & [\text{Holds}(\alpha, i) \wedge (i \text{ meets } j) \rightarrow \text{Holds}(\beta, j), \{i/(1, 2), j/(2, 5)\}]
 \end{aligned}$$

Given that we have a formula with a variable interval, we can consider the set of all possible groundings of that formula; to define this we use the complete grounding function.

**Definition 2.6.** The **complete grounding function**  $G_C(\Phi)$ , where  $\Phi$  is a set of  $\mathcal{T}$  formulae, is defined such that  $G_C(\Phi) = \bigcup_{\phi \in \Phi} G_C(\phi)$ , and  $G_C(\phi)$  for a formula  $\phi$  is defined such that  $G_C(\phi)$  is the set containing all complete groundings  $[\phi, \Theta]$ .

On a similar note, we have the zero grounding function, which converts a formula into a grounding, with no actual substitution done.

**Definition 2.7.** The **zero grounding function**  $G_0(\Phi)$ , where  $\Phi$  is a set of  $\mathcal{T}$  formulae, is defined such that  $G_0(\Phi) = \{[\phi, \emptyset] : \phi \in \Phi\}$ .

For convenience, we extend the definition of complete grounding to be applicable to entire sets.

**Definition 2.8.** The notation for the grounding function  $G_C(\Phi)$  can be extended to allow groundings of existing groundings. Specifically, we stipulate that, for the function  $G_C$  from definition 2.6 the following additional rules hold:

- For any grounding  $[\phi, \Theta]$ ,  $G_C([\phi, \Theta])$  is the set  $\{[\phi, \Theta'] : [\phi, \Theta'] \in G_C(\phi), \Theta \subseteq \Theta'\}$ . This is the set containing all complete groundings which are extensions of the existing grounding.
- For any set of groundings  $\Psi$ ,  $G_C(\Psi) = \bigcup_{\psi \in \Psi} G_C(\psi)$

Using this function, we can convert a set of formulae, or an entire database, into a set containing only completely ground formulae.

There are of course times when we wish to convert from a grounding back to a formula in  $\mathcal{T}$ , and we use the *Subst* function, below. This simply takes each entry in the substitution set, and replaces every occurrence of the variable with the fixed interval.

**Definition 2.9.** The **substitution function**,  $Subst([\alpha, \Theta])$ , where  $[\alpha, \Theta]$  is a grounding, returns a single formula  $\alpha'$ , such that, for each  $i/(a, b) \in \Theta$ , all occurrences of  $i$  in  $\alpha$  are replaced by  $(a, b)$  in  $\alpha'$

Some useful results concerning the complete grounding function can be observed, showing that the application of  $G_C$  does not cause any problems with set membership; these results prove useful in later proofs. Firstly, we can apply  $G_C$  repeatedly with no effect.

**Theorem 2.1.** For any set of  $\mathcal{T}$  formulae  $\Phi$ ,  $G_C(G_C(\Phi)) = G_C(G_0(\Phi)) = G_C(\Phi)$

Secondly, we can distribute  $\cup$  into  $G_C$ .

**Theorem 2.2.**  $G_C(\Phi) \cup G_C(\Psi) = G_C(\Phi \cup \Psi)$ .

And as a consequence of this, we can show that the subset relationship is maintained through complete grounding.

**Theorem 2.3.** Suppose  $\Phi \subseteq \Psi$ . Then  $G_C(\Phi) \subseteq G_C(\Psi)$ .

This demonstrates a useful property with respect to arguments: if  $\Phi$  is a minimal subset of a database  $\Delta$  such that  $\Phi$  entails  $\alpha$ , then we wish there to be a minimal subset  $\Psi \subseteq G_C(\Phi) \subseteq G_C(\Delta)$  such that  $\Psi$  also entails  $\alpha$ .

### 2.3. The consequence relation $\vdash_{\mathcal{T}}$

For the consequence relation in  $\mathcal{T}$ , we define a translation into a classical propositional language  $\mathcal{T}'$ , and then define  $\vdash_{\mathcal{T}}$  in terms of  $\vdash$  (i.e. the classical consequence relation). We also need to add some predefined knowledge about interval relationships into  $\vdash_{\mathcal{T}}$ , so that, for example,  $(0, 1)$  **meets**  $(1, 2)$  is a tautology.

**Definition 2.10.** The language  $\mathcal{T}'$  is a classical propositional language with the following syntax for a letter:

$$\begin{aligned} symbol &::= \alpha \mid \beta \mid \gamma \mid \delta \dots \\ index &::= 0 \mid 1 \mid 2 \mid 3 \dots \\ relation &::= \text{precedes} \mid \text{meets} \mid \text{overlaps} \mid \text{during} \\ &\quad \mid \text{starts} \mid \text{ends} \mid \text>equals \\ letter &::= symbol_{(index, index)} \\ &\quad \mid relation_{(index, index), (index, index)} \end{aligned}$$

Where *symbol* is any *symbol* in  $\mathcal{T}$ , and *index* is any *timepoint* in  $\mathcal{T}$  (and thus both are of finite size).

Combined with the usual connectives, it is fairly simple to define a function which converts from a completely ground formula in  $\mathcal{T}$  to  $\mathcal{T}'$ .

**Definition 2.11.** The function *Prop* applied to a complete grounding results in a formula in  $\mathcal{T}'$ , and can be defined recursively as below:

- $\text{Prop}([\alpha, \Theta]) = \text{PropSub}(\text{Subst}(\alpha, \Theta))$
- $\text{PropSub}(\text{Holds}(\alpha, (a, b))) = \alpha_{(a, b)}$
- For any relation (eg **meets**, from definition 2.10),  
 $\text{PropSub}((a, b) \text{ relation } (c, d)) = \text{relation}_{(a, b), (c, d)}$
- $\text{PropSub}(\neg\alpha) = \neg\text{PropSub}(\alpha)$
- $\text{PropSub}(\alpha \vee \beta) = \text{PropSub}(\alpha) \vee \text{PropSub}(\beta)$
- $\text{PropSub}(\alpha \wedge \beta) = \text{PropSub}(\alpha) \wedge \text{PropSub}(\beta)$

**Definition 2.12.** When *Prop* is applied to a set  $\Phi$  of groundings in  $\mathcal{T}$ , the result is as if *Prop* was applied to each member in turn, ie  $\text{Prop}(\Phi) = \bigcup_{\phi \in \Phi} \text{Prop}(\phi)$

The second part of  $\vdash_{\mathcal{T}}$  requires knowledge of which interval relationships are “true” and which are not. This is done by the set  $I$ , which contains this information.

**Definition 2.13.** The set  $I$  of interval relations is  $I^\top \cup I^\perp$ , where:

- $I^\top$  is the set of all formulae  $((a, b) \text{ relation } (c, d))$  such that  $(a, b)$  and  $(c, d)$  are intervals, **relation** is a relation from definition 2.3 and the conditions on  $a, b, c$  and  $d$  from definition 2.3 hold.
- $I^\perp$  is the set of all formulae  $\neg((a, b) \text{ relation } (c, d))$  such that  $(a, b)$  and  $(c, d)$  are intervals, **relation** is a relation from definition 2.3 and the conditions on  $a, b, c$  and  $d$  from definition 2.3 do not hold.

**Example 2.3.** The set  $I$  will contain formulae such as  $((0, 3) \text{ meets } (3, 5))$  (in  $I^\top$ ) and  $\neg((0, 7) \text{ before } (2, 4))$  (in  $I^\perp$ ).

With the *Prop* and  $I$ , we can define  $\vdash_{\mathcal{T}}$ .

**Definition 2.14.** Let  $\Phi$  be a set of formulae in  $\mathcal{T}$  and  $\alpha$  a formula in  $\mathcal{T}$ . Then  $\Phi \vdash_{\mathcal{T}} \alpha$  iff

$$\text{Prop}(G_C(\Phi) \cup I) \vdash \text{Prop}(\bigwedge G_C(\{\alpha\}))$$

where  $I$  is the set of interval relations given in definition 2.13 and  $\vdash$  is the classical consequence relation.

The use of  $\bigwedge G_C(\{\alpha\})$  here is necessary in order to allow the function *Prop* to be applied should  $\alpha$  contain intervals; effectively, we say that we can conclude *Holds(sales\_rising, i)* only if we can prove that *sales\_rising* holds for every possible interval  $(a, b)$ .

### 3. Arguments in $\mathcal{T}$

Before we begin, we make two simple assumptions about databases in  $\mathcal{T}$ . Firstly, for a database  $\Delta$  we assume that each subset of  $G_C(\Delta)$  is given an enumeration  $\langle \alpha_1, \dots, \alpha_n \rangle$ , known as the canonical enumeration. Also, in a database, each entry is assumed to have a unique numerical label, and using this label,  $(1)_{i=(0,1)}$  is shorthand for the database entry 1 ground with  $[i/(0, 1)]$ . (1) is shorthand for the zero grounding of database entry 1.

#### 3.1. Argument Definitions

We can alter the established definition of an argument (eg Amgoud and Cayrol [7], Besnard and Hunter [6]) to work with  $\mathcal{T}$  easily enough, with the definition of an unground argument below.

**Definition 3.1.** An **unground  $\mathcal{T}$  argument** (or just an unground argument) is a pair  $\langle \Phi, \alpha \rangle$ , assuming a database  $\Delta$ , such that:

- (1)  $\Phi \not\models_{\mathcal{T}} \perp$
- (2)  $\Phi \vdash_{\mathcal{T}} \alpha$
- (3)  $\Phi$  is a minimal subset of  $\Delta$  satisfying (1) and (2)

**Example 3.1.** Suppose we have the database (where *sales* denotes high sales):

- (1)  $Holds(sales, i) \wedge (j \text{ during } i) \rightarrow Holds(sales, j)$
- (2)  $Holds(sales, (0, 8))$

Then there is an unground argument  $\langle \{(1), (2)\}, Holds(sales, (3, 4)) \rangle$

The example above is a case where we can exhibit the characteristics of Shoham's downward hereditary propositions [15], in that if we have high sales over an interval, we can say that we have high sales in the subintervals of that interval.

However, as is indicated by the name unground above, there are alternate approaches to the definition of an argument we can use, such as the definition of a database ground argument below, so called because the database itself is ground before the argument is taken.

**Definition 3.2.** A **database ground  $\mathcal{T}$  argument** (or simply database ground argument) is a pair  $\langle \Phi, \alpha \rangle$ , assuming a database  $\Delta$ , such that:

- (1)  $\Phi \not\models_{\mathcal{T}} \perp$
- (2)  $\Phi \vdash_{\mathcal{T}} \alpha$
- (3)  $\Phi$  is a minimal subset of  $G_C(\Delta)$  satisfying (1) and (2)

This gives two advantages, the first simply being that of a more informative support, since we can state exactly which intervals we are using in place of the variables in the support.

**Example 3.2.** Using the database of example 3.1, we have the following database ground argument:

$$\langle \{(1)_{i=(0,8),j=(3,4)}, (2)\}, Holds(sales, (3, 4)) \rangle$$

However, there is another possible advantage over the simpler definition above, as in the example below, where we can produce arguments that are not possible with the above definition.

**Example 3.3.** Suppose we have the database:

- (1)  $Holds(sales, i) \rightarrow Holds(profits, i)$
- (2)  $Holds(sales, (0, 1)) \wedge Holds(sales(1, 2)) \wedge \neg Holds(profits(1, 2))$

Then, despite the fact that (1) and (2) are mutually inconsistent, there is a database ground argument  $\langle \{(1)_{i=(0,1)}, (2)\}, Holds(profits, (0, 1)) \rangle$

This example is perhaps best considered with respect to news reports; the first formula in the database would be part of background knowledge, and the second a news report saying that a company has had good sales last quarter and this quarter, but profits appear to be down.

We could also use a different grounding function; the support ground argument (so called because the support is ground) uses the zero grounding function.

**Definition 3.3.** A **support ground  $\mathcal{T}$  argument** (or just support ground argument) is a pair  $\langle \Phi, \alpha \rangle$ , assuming a database  $\Delta$ , such that:

- (1)  $\Phi \not\vdash_{\mathcal{T}} \perp$
- (2)  $\Phi \vdash_{\mathcal{T}} \alpha$
- (3)  $\Phi$  is a minimal subset of  $G_0(\Delta)$  satisfying (1) and (2)

**Example 3.4.** Suppose we have the database:

- (1)  $Holds(sales, i) \rightarrow Holds(profits, i)$
- (2)  $Holds(sales, (0, 1))$

Then there is a support ground argument  $\langle \{(1), (2)\}, Holds(profits, (0, 1)) \rangle$

In fact, a support ground argument is no different from an unground argument, as formalized in the theorem below. Nevertheless, we use support ground arguments rather than unground arguments as a comparison to database ground arguments, simply because the support of both support and database ground arguments consists of a set of groundings, and so they are more directly comparable.

**Theorem 3.1.**  $\langle \Psi, \alpha \rangle$  is a support ground argument iff  $\langle \Phi, \alpha \rangle$  is an unground argument, where  $\Psi$  is the set  $\{[\phi, \emptyset] : \phi \in \Phi\}$

While there are obvious similarities between the above definitions of an argument, there is a tangible difference between database ground and support ground arguments, as in the examples above; in example 3.3 there is no support ground argument. However, we can show that that there are always at least as many database ground arguments as support ground ones.

**Theorem 3.2.** Given a finite database  $\Delta$ , for every support ground argument  $\langle \Phi, \alpha \rangle$ , there is at least one database ground argument  $\langle \Psi, \alpha \rangle$ , such that  $\Psi \subseteq G_C(\Phi)$

### 3.2. Conservatism and Canonical Undercuts

The following section, and the theorems therein are adapted from Besnard and Hunter's argumentation system [6], and indeed there is little difference; the addition of either grounding function does not affect conservatism.

Conservatism is a way of finding relationships between arguments, and is used to reduce the number of similar arguments we consider when making a point; a more conservative argument has a less specific conclusion and/or uses less information. The aim of conservatism and canonical undercuts is to reduce the number of similar but valid arguments we can use to attack another argument; for  $\langle \{\alpha, \alpha \leftrightarrow \beta\}, \beta \rangle$  we could attack with both  $\langle \{\neg\alpha\}, \neg(\alpha \wedge \alpha \leftrightarrow \beta) \rangle$  and  $\langle \{\neg\alpha, \alpha \leftrightarrow \beta\}, \neg\beta \rangle$ , but these effectively make the same point in their attack, so we prefer one over the other (in this case, the former over the latter).

**Definition 3.4.** A  $\mathcal{T}$  argument  $\langle \Phi, \alpha \rangle$  is **more conservative than** a  $\mathcal{T}$  argument  $\langle \Psi, \beta \rangle$  iff  $\Phi \subseteq \Psi$  and  $\beta \vdash_{\mathcal{T}} \alpha$ .

**Theorem 3.3.** Being more conservative forms a pre-order on arguments in  $\mathcal{T}$ . Maximally conservative arguments exist, these are  $\langle \emptyset, \top \rangle$ , where  $\top$  is any tautology.

Undercuts are arguments which undercut the support of another argument by contradicting some part of their support.

**Definition 3.5.** An **undercut** for a  $\mathcal{T}$  argument  $\langle \Phi, \alpha \rangle$  is a  $\mathcal{T}$  argument  $\langle \Psi, \neg(\phi_1 \wedge \dots \wedge \phi_n) \rangle$  where  $\{\phi_1, \dots, \phi_n\} \subseteq \Phi$ .

**Example 3.5.** Given the formula

$$(3) \quad \text{Holds}(\text{sales}(1, 2)) \wedge \neg\text{Holds}(\text{profits}(1, 2))$$

Then  $\langle \{(3)\}, \neg(\text{Holds}(\text{sales}, i) \rightarrow \text{Holds}(\text{profits}, i)) \rangle$  is an undercut for the support ground argument of example 3.4. Note this is not an undercut for the database ground argument of example 3.3.

While other types of defeater exist, such as the rebuttal and the more general defeater, undercuts can be shown, as in Besnard and Hunter's paper [6], to be representative of other defeaters, and so in this paper we simply consider undercuts.

**Definition 3.6.** A  $\mathcal{T}$  argument  $\langle \Psi, \beta \rangle$  is a **maximally conservative undercut** for another  $\mathcal{T}$  argument  $\langle \Phi, \alpha \rangle$  iff  $\langle \Psi, \beta \rangle$  is a undercut of  $\langle \Phi, \alpha \rangle$  such that there is no undercut of  $\langle \Phi, \alpha \rangle$  which is strictly more conservative than  $\langle \Psi, \beta \rangle$ .

While maximally conservative undercuts are a good representative of many similar undercuts, a canonical undercut solves the problem of both  $\langle \{\neg\alpha, \neg\beta\}, \neg(\alpha \wedge \beta) \rangle$  and  $\langle \{\neg\alpha, \neg\beta\}, \neg(\beta \wedge \alpha) \rangle$  being maximally conservative.

**Definition 3.7.** A **canonical undercut** for a  $\mathcal{T}$  argument  $\langle \Phi, \alpha \rangle$  is a maximally conservative  $\mathcal{T}$  undercut  $\langle \Psi, \neg(\phi_1 \wedge \dots \wedge \phi_n) \rangle$  such that  $\langle \phi_1 \dots \phi_n \rangle$  is the canonical enumeration of  $\Phi$ .

**Example 3.6.** Continuing example 3.5, the following is a canonical undercut for the support ground argument of example 3.4:

$$\langle\{(3)\}, \neg(Holds(sales, (0, 1)) \wedge (Holds(sales, i) \rightarrow Holds(profits, i)))\rangle$$

Given the fact that the conclusion of a canonical undercut is always the negation of the conjunction of the support (shown in [6]), we use the symbol  $\diamond$  in place of this, as a shorthand.

#### 4. Argument Trees

As in Besnard and Hunter's argumentation system [6,8], we can put sets of arguments into argument trees, according to the definition below.

**Definition 4.1.** An **argument tree** for  $\alpha$  is a tree where the nodes are  $T$  arguments, such that:

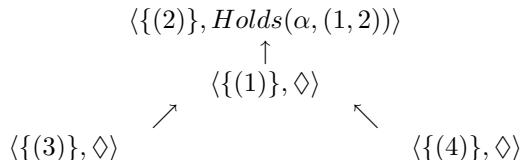
- (1) The root node is an argument for  $\alpha$
- (2) For no node  $\langle\Phi, \beta\rangle$  with ancestor nodes  $\langle\Phi_1, \beta_1\rangle \dots \langle\Phi_n, \beta_n\rangle$  is  $\Phi$  a subset of  $\Phi_1 \cup \dots \cup \Phi_n$ .
- (3) The children of a node  $N$  consist of all canonical undercuts which obey (2).

While argument trees may seem superficially similar for database ground and support ground arguments, and indeed the definition is identical apart from the type of  $T$  argument, the trees themselves are not always similar, and the example below sums up many of the differences in both shape and size between database ground and support ground argument trees.

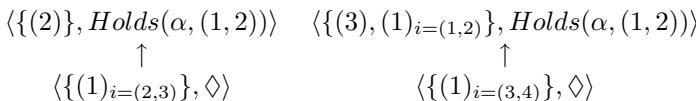
**Example 4.1.** Considering the following database:

- (1)  $Holds(\beta, i) \rightarrow Holds(\alpha, i)$
- (2)  $Holds(\alpha, (1, 2)) \wedge Holds(\beta, (2, 3)) \wedge \neg Holds(\alpha, (2, 3))$
- (3)  $Holds(\beta, (1, 2)) \wedge Holds(\beta, (3, 4)) \wedge \neg Holds(\alpha, (3, 4))$
- (4)  $Holds(\beta, (5, 6)) \wedge \neg Holds(\alpha, (5, 6))$

Suppose we wish to argue for  $Holds(\alpha, (1, 2))$ . We have the following support ground argument tree:



We also have the following pair of database ground argument trees, the first with a root corresponding to the support ground argument above, and the second having a root with no corresponding support ground argument.



There is a very limited relationship between these trees; since for every database ground argument there is a support ground argument, we know that there is at least one database ground tree with the same root (or a related root) for each support ground tree. We can take this slightly further, with the theorem below.

**Theorem 4.1.** For every support ground argument tree, there is a database ground argument tree, such that for every branch  $\langle \Phi_1, \alpha \rangle, \langle \Phi_2, \diamond \rangle, \dots, \langle \Phi_m, \diamond \rangle$  in the support ground argument tree, there is a branch in the database ground argument tree  $\langle \Psi_1, \alpha \rangle, \langle \Psi_2, \diamond \rangle, \dots, \langle \Psi_n, \diamond \rangle$  (where  $n > 0$ ) such that, for some  $p$  where  $0 \leq p \leq m$  and  $p \leq n$ , the following holds:

- (1)  $\Phi_p$  is the last support (if any) in  $\Phi_1 \dots \Phi_m$  such that, for all  $i$  such that  $0 < i \leq p$ ,  $\Phi_i$  is a set of only completely ground formulae.
- (2) For all  $i$  such that  $1 \leq i \leq p$ ,  $\Psi_i = \Phi_i$ .
- (3) if  $p < n$  and  $p < m$ , then  $\Psi_{p+1} \subseteq G_C(\Phi_{p+1})$ .

Note that because of theorem 3.2, it follows directly that the number of trees with the same conclusion in the support ground case can never exceed the number of database ground trees.

Unfortunately, while database ground arguments have their advantages, they do have additional problems. Consider the following example:

**Example 4.2.** Consider the database:

- (1)  $Holds(\alpha, i)$
- (2)  $Holds(\beta, i)$
- (3)  $Holds(\beta, i) \rightarrow \neg Holds(\alpha, i)$

In this database, suppose we form arguments with the conclusion  $Holds(\alpha, i)$ . The support ground argument tree is very simple:

$$\begin{array}{c} \langle \{(1)\}, Holds(\alpha, i) \rangle \\ \uparrow \\ \langle \{(2), (3)\}, \diamond \rangle \end{array}$$

However, the database ground argument tree has a very large number of leaves; in fact the leaf nodes are all nodes of the form  $\langle \{(2)_{i=(a,b)}, (3)_{i=(a,b)}\}, \diamond \rangle$  where  $a \in T, b \in T, a \prec b$  and  $T$  is the set of timepoints; part of this tree is as below. The syntax  $G_C((1))$  is used to represent the set containing every possible complete grounding of (1).

$$\begin{array}{ccccc} & \langle G_C((1)), Holds(\alpha, i) \rangle & & & \\ & \nearrow & \uparrow & & \dots \\ \langle \{(2)_{i=(1,2)}, (3)_{i=(1,2)}\}, \diamond \rangle & & \langle \{(2)_{i=(2,3)}, (3)_{i=(2,3)}\}, \diamond \rangle & & \dots \end{array}$$

Here, we can see an example of a tree with a potentially infinite width (it is not actually infinitely wide, since there are a finite number of timepoints, and so a finite number of possible groundings). This is clearly an undesirable feature, and further work, beyond the scope of this paper, examines possible solutions to problems such as these.

## 5. Discussion

In this paper we have discussed a way of encoding temporal information into propositional logic, and examined its impact in a coherence argumentation system.

We have shown that it is possible to use the standard definition of an argument, but there are other possible argument definitions that arise through the use of grounding in the definition of  $\vdash_T$ , the database ground and support ground arguments, and we have shown that while the definitions of these are similar, there are differences, in particular in the case of situations such as example 3.3 where there are no support ground arguments.

Although theorem 4.1 shows there is some comparison between the support ground and database ground cases, when we put these arguments into the argument trees of Besnard and Hunter, there are some significant differences, including different depths and widths of tree or different numbers of trees.

While database ground arguments and support ground/unground arguments both have their advantages, we do not consider that either is the “best” type, and each has their own uses. In particular, example 4.2 shows that while there are benefits that database ground arguments may provide over support ground or unground arguments, they also have disadvantages. Nevertheless, we consider that, particularly with further work, the use of grounding within the argument definition provides definite advantages and a useful avenue of study for further work in temporal argumentation.

## References

- [1] H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In D. Gabbay, editor, *Handbook of Philosophical Logic*. Kluwer, 2000.
- [2] C. I. Chesñevar, A. Maguitman, and R. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.
- [3] Ph. Besnard and A. Hunter. *Elements of Argumentation*. MIT Press, 2008.
- [4] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
- [5] A. Garcia and G. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.
- [6] Ph. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128:203–235, 2001.
- [7] L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–216, 2002.
- [8] Ph. Besnard and A. Hunter. Practical first order argumentation. In *Proceedings of the 20th American National Conference on Artificial Intelligence (AAAI'2005)*. MIT Press, 2005.
- [9] R. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing*, 4:67–95, 1986.
- [10] A. Prior. *Past, Present and Future*. Clarendon Press, Oxford, 1967.
- [11] J. F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.
- [12] A. Hunter. Ramification analysis with structured news reports using temporal argumentation. In *Proceedings of the Adventures in Argumentation Workshop (part of the Sixth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty)*. Institut de Recherche en Informatique de Toulouse, 2001.
- [13] J. C. Augusto and G. Simari. A temporal argumentation system. *AI Communications*, 12(4):237–257, 1999.
- [14] C. Hamblin. Instants and intervals. In J. F. H. Fraser and G. Muller, editors, *The Study of Time*, pages 324–328. Springer, 1972.
- [15] Y. Shoham. Temporal logics in ai: Semantic and ontological considerations. *Artificial Intelligence*, 33:89–104, 1987.

# Strong and Weak Forms of Abstract Argument Defense

Diego C. MARTÍNEZ Alejandro J. GARCÍA Guillermo R. SIMARI

*Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET),*

*Artificial Intelligence Research and Development Laboratory LIDIA,*

*Department of Computer Science and Engineering, Universidad Nacional del Sur;*

*Bahía Blanca, ARGENTINA*

*Email: {dcm,ajg,grs}@cs.uns.edu.ar*

**Abstract.** Extended abstract frameworks separate conflicts and preference between arguments. These elements are combined to induce argument defeat relations. A proper defeat is consequence of preferring an argument in a conflicting pair, while blocking defeat is consequence of incomparable or equivalent-in-strength conflicting arguments. As arguments interact with different strengths, the quality of several argument extensions may be measured in a particular semantics. In this paper we analyze the strength of defenses in extended argumentation frameworks, under admissibility semantics. A more flexible form of acceptability is defined leading to a credulous position of acceptance.

**Keywords.** Abstract argumentation, admissibility semantics, credulous acceptance.

## 1. Introduction

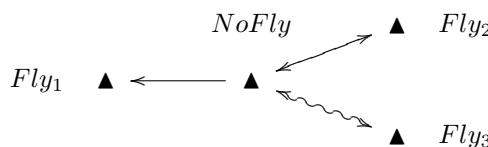
Abstract argumentation systems [9,14,1,2] are formalisms for argumentation where some components remain unspecified towards the study of pure semantic notions. Most of the existing proposals are based on the single abstract concept of *attack* represented as a binary relation, and according to several rational rules, extensions are defined as sets of possibly accepted arguments. The attack relation is basically a subordinate relation of conflicting arguments. For two arguments  $\mathcal{A}$  and  $\mathcal{B}$ , if  $(\mathcal{A}, \mathcal{B})$  is in the attack relation, then the status of acceptance of  $\mathcal{B}$  is conditioned by the status of  $\mathcal{A}$ , but not the other way around. It is said that argument  $\mathcal{A}$  attacks  $\mathcal{B}$ , and it implies a priority between conflicting arguments. It is widely understood that this priority is related to some form of argument strength. This is easily modeled through a preference order. Several frameworks do include an argument order [1,3,5], although the classic attack relation is kept, and therefore this order is not used to induce attacks.

In [11,10] an extended abstract argumentation framework is introduced, where two kinds of defeat relations are present. These relations are obtained by applying a preference criterion between conflictive arguments. The conflict relation is kept in its most basic, abstract form: two arguments are in conflict simply if both arguments cannot be accepted simultaneously. The preference criterion subsumes any evaluation on arguments

and it is used to determine the direction of the attack. Consider the following arguments, showing a simple argument defense:

- $Fly_1$ : Fly Oceanic Airlines because it has the cheapest tickets.
- $NoFly$ : Do not fly Oceanic Airlines because the accident rate is high and the onboard service is not good.
- $Fly_2$ : Fly Oceanic Airlines because the accident rate is normal and the onboard service is improving.
- $Fly_3$ : Fly Oceanic Airlines because you can see some islands in the flight route.

There is a conflict between  $NoFly$  and the other arguments. This conflict is basically an acceptance constraint: if  $NoFly$  is accepted, then  $Fly_1$  should be rejected (as well as  $Fly_2$  and  $Fly_3$ ) and viceversa. A comparison criterion may be applied towards a decision for argument acceptance. Suppose the following preferences are established. Argument  $NoFly$  is considered a stronger argument than  $Fly_1$ , as it includes information about accidents and service, which is more important than ticket fares. Then  $NoFly$  is said to be a *proper defeater* of  $Fly_1$ . Argument  $Fly_2$  is also referring to safety and service, and therefore is considered as strong as  $NoFly$ . Both arguments are blocking each other and thus they are said to be *blocking defeaters*. Argument  $Fly_3$  is referring to in-flight sightseeing, which is not related to the topics addressed by  $NoFly$ . Both arguments are incomparable to each other, and they are considered blocking defeaters. This situation can be depicted using graphs, with different types of arcs. Arguments are represented as black triangles. An arrow ( $\rightarrow$ ) is used to denote proper defeaters. A double-pointed straight arrow ( $\leftrightarrow$ ) connects blocking defeaters considered equivalent in strength, and a double-pointed zig-zag arrow ( $\rightsquigarrow$ ) connects incomparable blocking defeaters. In Figure 1, the previous arguments and its relations are shown. Argument  $NoFly$  is a proper defeater of  $Fly_1$ . Arguments  $Fly_2$  and  $NoFly$  are equivalent-in-strength blocking defeaters, and arguments  $Fly_3$  and  $NoFly$  are incomparable blocking defeaters.



**Figure 1.** EAF-graph

In this example, arguments  $Fly_2$  and  $Fly_3$  are defeaters of  $NoFly$ , which is a defeater of  $Fly_1$ . Therefore,  $Fly_1$  is said to be defended by  $Fly_2$  and  $Fly_3$ . Note that the best defense should be provided by an argument considered stronger than  $NoFly$ . Here, defense is achieved by blocking defeaters, that is, by arguments in symmetric opposition. Even then, the defenses are of different quality. The defense provided by  $Fly_2$  may be considered stronger than the one provided by  $Fly_3$ , as it is possible to compare  $Fly_2$  to  $NoFly$ , although just to conclude they have equivalent strength. The argument  $Fly_3$  cannot be compared to  $NoFly$  and then the conflict remains unevaluated. Incomparability arise because of incomplete information, or because the arguments are actually incomparable. This may be viewed as the weakest form of defense. These different grades of defense, achieved as a result of argument comparison, is the main motivation of this work.

In this paper we formalize the strength of defenses for arguments and we explore its relation to classic semantic notions. This paper is organized as follows. In the next section, the extended argumentation frameworks are formally introduced. In Section 3 the notion of admissibility is applied to extended frameworks. In Section 4, the strength of defenses is formalized and this notion is applied to analyze the grounded extension in Section 5. In order to adopt a credulous position, a more flexible form of acceptability is defined in Section 6. Finally, the conclusions are presented in Section 7.

## 2. Extended Abstract Frameworks

In our extended argumentation framework three relations are considered: *conflict*, *subargument* and *preference* between arguments. The definition follows:

**Definition 1 (Extended Framework)** *An extended abstract argumentation framework (called EAF) is a quartet  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$ , where  $\text{Args}$  is a finite set of arguments, and  $\sqsubseteq$ ,  $\mathbf{C}$  and  $\mathbf{R}$  are binary relations over  $\text{Args}$  denoting respectively subarguments, conflicts and preferences between arguments.*

Arguments are abstract entities, as in [9], that will be denoted using calligraphic uppercase letters, possibly with indexes. In this work, the subargument relation is not relevant for the topic addressed. Basically, it is used to model the fact that arguments may include inner pieces of reasoning that can be considered arguments by itself, and it is of special interest in dialectical studies [12]. Hence, unless explicitly specified, in the rest of the paper  $\sqsubseteq = \emptyset$ . The conflict relation  $\mathbf{C}$  states the incompatibility of acceptance between arguments, and thus it is a symmetric relation. Given a set of arguments  $S$ , an argument  $\mathcal{A} \in S$  is said to be in conflict in  $S$  if there is an argument  $\mathcal{B} \in S$  such that  $\{\mathcal{A}, \mathcal{B}\} \in \mathbf{C}$ . The relation  $\mathbf{R}$  is introduced in the framework and it will be used to evaluate arguments, modeling a preference criterion based on a measure of strength.

**Definition 2 (Comparison criterion)** *Given a set of arguments  $\text{Args}$ , an argument comparison criterion  $\mathbf{R}$  is a binary relation on  $\text{Args}$ . If  $\mathcal{A} \mathbf{R} \mathcal{B}$  but not  $\mathcal{B} \mathbf{R} \mathcal{A}$  then  $\mathcal{A}$  is strictly preferred to  $\mathcal{B}$ , denoted  $\mathcal{A} > \mathcal{B}$ . If  $\mathcal{A} \mathbf{R} \mathcal{B}$  and  $\mathcal{B} \mathbf{R} \mathcal{A}$  then  $\mathcal{A}$  and  $\mathcal{B}$  are indifferent arguments with equal relative preference, denoted  $\mathcal{A} \equiv \mathcal{B}$ . If neither  $\mathcal{A} \mathbf{R} \mathcal{B}$  or  $\mathcal{B} \mathbf{R} \mathcal{A}$  then  $\mathcal{A}$  and  $\mathcal{B}$  are incomparable arguments, denoted  $\mathcal{A} \bowtie \mathcal{B}$ .*

For two arguments  $\mathcal{A}$  and  $\mathcal{B}$  in  $\text{Args}$ , such that the pair  $\{\mathcal{A}, \mathcal{B}\}$  belongs to  $\mathbf{C}$  the relation  $\mathbf{R}$  is considered, in order to elucidate the conflict. Depending on the preference order, two main notions of argument defeat are derived.

**Definition 3 (Defeaters)** *Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF and let  $\mathcal{A}$  and  $\mathcal{B}$  be two arguments such that  $(\mathcal{A}, \mathcal{B}) \in \mathbf{C}$ . If  $\mathcal{A} > \mathcal{B}$  then it is said that  $\mathcal{A}$  is a proper defeater of  $\mathcal{B}$ . If  $\mathcal{A} \equiv \mathcal{B}$  or  $\mathcal{A} \bowtie \mathcal{B}$ , it is said that  $\mathcal{A}$  is a blocking defeater of  $\mathcal{B}$ , and viceversa. An argument  $\mathcal{B}$  is said to be a defeater of an argument  $\mathcal{A}$  if  $\mathcal{B}$  is a blocking or a proper defeater of  $\mathcal{A}$ .*

**Example 1** Let  $\Phi_1 = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF where  $\text{Args} = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}\}$ ,  $\sqsubseteq = \emptyset$ ,  $\mathbf{C} = \{\{\mathcal{A}, \mathcal{B}\}, \{\mathcal{B}, \mathcal{C}\}, \{\mathcal{C}, \mathcal{D}\}, \{\mathcal{C}, \mathcal{E}\}\}$  and  $\mathcal{A} > \mathcal{B}, \mathcal{B} > \mathcal{C}, \mathcal{E} \bowtie \mathcal{C}, \mathcal{C} \equiv \mathcal{D}$ . Argument  $\mathcal{A}$  is a proper defeater of  $\mathcal{B}$ , arguments  $\mathcal{E}$  and  $\mathcal{C}$  are blocking each other. The same is true for  $\mathcal{C}$  and  $\mathcal{D}$ .

Several semantic notions were defined for EAFs. In [10], a set of accepted arguments is captured through a fixpoint operator based on the identification of *suppressed arguments*. In [12], an extension based on proof restrictions (*progressive argumentation lines*) is introduced. Semantic notions regarding strength of defenses are discussed in [13], which leads to this present work.

In the next section, the classic acceptability notion is applied to extended abstract frameworks, in order to analyze the composition of inner defenses.

### 3. Admissibility

Argumentation semantics is about argument classification through several rational positions of acceptance. A central notion in most argument extensions is *acceptability*. A very simple definition of acceptability in extended abstract frameworks is as follows.

**Definition 4 (Acceptability in EAF)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. An argument  $\mathcal{A} \in \text{Args}$  is acceptable with respect to a set of arguments  $S \subseteq \text{Args}$  if and only if every defeater  $\mathcal{B}$  of  $\mathcal{A}$  has a defeater in  $S$ .

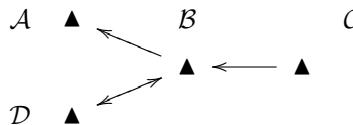


Figure 2. Simple extended abstract framework

Defeaters mentioned in Definition 4 may be either proper or blocking ones. In the Figure 2, argument  $\mathcal{A}$  is acceptable with respect to  $\{\mathcal{C}\}, \{\mathcal{D}\}$  and of course  $\{\mathcal{C}, \mathcal{D}\}$ . Argument  $\mathcal{C}$  is acceptable with respect to the empty set. Argument  $\mathcal{D}$  is acceptable with respect to  $\{\mathcal{C}\}$  and also with respect to  $\{\mathcal{D}\}$ .

Following the usual steps in argumentation semantics, the notion of acceptability leads to the notion of admissibility. This requires the definition of conflict-free set of arguments. A set of arguments  $S \subseteq \text{Args}$  is said to be *conflict-free* if for all  $\mathcal{A}, \mathcal{B} \in S$  it is not the case that  $\{\mathcal{A}, \mathcal{B}\} \in \mathbf{C}$ .

**Definition 5 (Admissible set)** [9] Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. A set of arguments  $S \subseteq \text{Args}$  is said to be *admissible* if it is conflict-free and every argument in  $S$  is acceptable with respect to  $S$ .

An admissible set is able to defend any argument included in that set. Note that any blocked argument has an intrinsic self-defense due to the symmetry of blocking defeat. In [13] a restricted notion of admissibility is presented, called *x-admissibility*, where an argument is required to be defended by other arguments. Arguments that cannot be defended but by themselves are not included in x-admissible extensions.

#### 4. Defenses and strength

An argument  $\mathcal{A}$  may be collectively defended by several sets of arguments. The final set of accepted arguments depends on the position adopted by the rational agent, for which the knowledge is modeled by the framework. Several extensions are proposed to address this issue, as in [7,8,4,6]. An interesting position, suitable for EAFs, is to focus the acceptance in sets of strongly defended arguments. This corresponds to an agent that, given a special situation where arguments are defended in different manners, decides to accept a set where, if possible, the strongest defense is available. This is of special interest when a lot of equivalent-in-strength or incomparable arguments are involved in the defeat scenario.

This evaluation of defenses can be achieved by considering a rational, implicit ordering of argument preferences. We previously suggested this order, stating that the best defense is achieved through proper defeaters. If this is not the case, then at least is desirable a defense strong enough to block the attacks, and this is done at best by equivalent-in-strength defeaters. The worst case is to realize that both arguments, the attacker and the defender, are not related enough to evaluate a difference in force, leading to the weakest form of defeat, where the basis is only the underlying conflict. This is formalized in the next definition.

**Definition 6** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $\mathcal{A}$  and  $\mathcal{B}$  be two arguments in  $\text{Args}$ . The function  $\text{pref} : \text{Args} \times \text{Args} \rightarrow \{0, 1, 2\}$  is defined as follows

$$\text{pref}(\mathcal{A}, \mathcal{B}) = \begin{cases} 0 & \text{if } \mathcal{A} \bowtie \mathcal{B} \\ 1 & \text{if } \mathcal{A} \equiv \mathcal{B} \\ 2 & \text{if } \mathcal{A} \succ \mathcal{B} \end{cases}$$

Definition 6 serves as a mean to compare individual defenses. For an argument  $\mathcal{A}$ , the strength of its defenders is evaluated as stated in the following definition.

**Definition 7 (Defender's strength)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $\mathcal{A} \in \text{Args}$  be an argument with defeater  $\mathcal{B}$ , which is defeated, in turn, by arguments  $\mathcal{C}$  and  $\mathcal{D}$ . Then

1.  $\mathcal{C}$  and  $\mathcal{D}$  are equivalent in force defenders of  $\mathcal{A}$  if  $\text{pref}(\mathcal{C}, \mathcal{B}) = \text{pref}(\mathcal{D}, \mathcal{B})$ .
2.  $\mathcal{C}$  is a stronger defender than  $\mathcal{D}$  if  $\text{pref}(\mathcal{C}, \mathcal{B}) > \text{pref}(\mathcal{D}, \mathcal{B})$ . It is also said that  $\mathcal{D}$  is a weaker defender than  $\mathcal{C}$ .

In the airline example, the argument  $Fly_2$  is a stronger defender of  $Fly_1$  than  $Fly_3$ . The evaluation of a collective defense follows from Definition 7, considering set of arguments acting as defenders.

**Definition 8 (Stronger Defense)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $\mathcal{A} \in \text{Args}$  be an argument acceptable with respect to  $S_1 \subseteq \text{Args}$ . A set of arguments  $S_2 \subseteq \text{Args}$  is said to be a stronger collective defense of  $\mathcal{A}$  if  $\mathcal{A}$  is acceptable with respect to  $S_2$ , and

1. There are no two arguments  $\mathcal{X} \in S_1$  and  $\mathcal{Y} \in S_2$  such that  $\mathcal{X}$  constitutes a stronger defense than  $\mathcal{Y}$
2. For at least one defender  $\mathcal{X} \in S_1$  of  $\mathcal{A}$ , there exists an argument  $\mathcal{Y} \in S_2 - S_1$  which constitutes a stronger defense of  $\mathcal{A}$ .

A set of arguments  $S_2$  is a stronger collective defense of  $\mathcal{A}$  than the set  $S_1$  if the force of defense achieved by elements in  $S_2$  is stronger than those in  $S_1$  in at most one defender, being the rest equivalent in force. Thus, every argument in  $S_1$  has a correlative in  $S_2$  that is a stronger or equivalent in force defender. The improvement in defense must occur through at least one new argument.

The strength of a defense is a pathway to analyze the structure of admissible sets. In extended abstract frameworks, defeat may occur in different ways, according to preference  $\mathbf{R}$ , and this can be used to evaluate the inner composition of an admissible set.

**Example 2** Consider the EAF of Figure 3. The admissible sets are  $\emptyset$  (trivial), every singleton set,  $\{\mathcal{A}, \mathcal{D}\}$  and  $\{\mathcal{A}, \mathcal{C}\}$ . Argument  $\mathcal{A}$  is defended by sets  $\{\mathcal{D}\}$  and  $\{\mathcal{C}\}$ , but the first one is a stronger collective defense than the second one. Then  $\{\mathcal{A}, \mathcal{D}\}$  is an admissible set with stronger inner defenses than  $\{\mathcal{A}, \mathcal{C}\}$

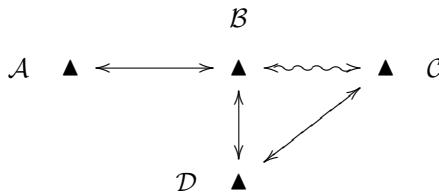


Figure 3.  $\{\mathcal{D}\}$  is a stronger defense of  $\mathcal{A}$  than  $\{\mathcal{C}\}$ .

The notion of strong admissible sets regarding inner defenses is captured in the following definition.

**Definition 9 (Top-admissibility)** An admissible set of arguments  $S$  is said to be top-admissible if, for any argument  $\mathcal{A} \in S$ , no other admissible set  $S'$  includes a stronger defense of  $\mathcal{A}$  than  $S$ .

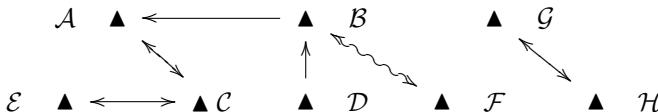


Figure 4. Argument defense

What top-admissibility semantics requires is that best admissible sets are selected, according to the strength of every defense. In Figure 4, the set  $\{\mathcal{A}, \mathcal{D}, \mathcal{E}\}$  is top-admissible, while the set  $\{\mathcal{A}, \mathcal{E}, \mathcal{F}\}$  is not. The sets  $\{\mathcal{G}\}$  and  $\{\mathcal{H}\}$  are also top-admissible. In the EAF of Figure 3, the set  $\{\mathcal{A}, \mathcal{D}\}$  is top-admissible.

The formalization of strong defended arguments needs the identification of the precise set of defenders. This is called an *adjusted defense*.

**Definition 10 (Adjusted defense)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $S$  be a set of arguments and let  $\mathcal{A} \in \text{Args}$  be an argument acceptable with respect to  $S$ . An argument  $\mathcal{B} \in S$  is a superfluous defender of  $\mathcal{A}$  in  $S$ , if  $\mathcal{A}$  is acceptable with respect to

$S - \{\mathcal{B}\}$ . If no argument in  $S$  is a superfluous-defender, then the set  $S$  is said to be an adjusted defense of  $\mathcal{A}$ .

In Figure 4, argument  $\mathcal{A}$  is acceptable with respect to, for instance,  $S = \{\mathcal{A}, \mathcal{D}, \mathcal{F}\}$ . However,  $S$  is not an adjusted defense of  $\mathcal{A}$ , as this argument is also acceptable with respect to subsets  $S_1 = \{\mathcal{A}, \mathcal{D}\}$  and  $S_2 = \{\mathcal{A}, \mathcal{F}\}$ , being these sets adjusted defenses for  $\mathcal{A}$ . As it is free of defeaters, argument  $\mathcal{D}$  has the empty set as the only adjusted defense, while the set  $\{\mathcal{G}\}$  is an adjusted defense of  $\mathcal{G}$ . The same is true for  $\mathcal{H}$ .

Note that it is possible for an argument to have more than one adjusted defense. In fact, if an argument  $\mathcal{X}$  is involved in a blocking defeat, it can be included in an adjusted defense for  $\mathcal{X}$ , since it is able to defend itself against non-preferred defeaters.

**Definition 11 (Dead-end defeater)** An argument  $\mathcal{B}$  is said to be a dead-end defeater of an argument  $\mathcal{A}$  if the only defense of  $\mathcal{A}$  against  $\mathcal{B}$  is  $\mathcal{A}$  itself.

Dead-end defeaters arise only when an argument  $\mathcal{A}$  is involved in a blocking situation with another argument without third defeaters, and therefore  $\mathcal{A}$  cannot appeal to other defenses on that attack. This leads to self-defender arguments.

**Definition 12 (Weak-acceptable)** An argument  $\mathcal{A}$  is said to be a self-defender if for every adjusted defense  $S$  of  $\mathcal{A}$ , then  $\mathcal{A} \in S$ . In that case,  $\mathcal{A}$  is said to be weak-acceptable with respect to  $S$  if (i)  $|S| > 1$ , and (ii)  $\mathcal{A}$  is defended by  $S - \{\mathcal{A}\}$  against every non dead-end defeater.

A self-defender argument  $\mathcal{A}$  is considered weak-acceptable with respect to a set  $S$  if it is actually defended by  $S$  from other attacks. If  $|S| > 1$  then clearly  $\mathcal{A}$  cannot be defended by itself against all its defeaters. Weak-acceptability requires  $\mathcal{A}$  to be defended whenever other defenders are available. This is because  $\mathcal{A}$ , being enforced to be a self-defeater, may be also defending itself against non dead-end defeaters.

In Figure 4, arguments  $\mathcal{G}$  and  $\mathcal{H}$  are self-defender arguments. Argument  $\mathcal{C}$  is weak-acceptable with respect to  $\{\mathcal{C}, \mathcal{B}\}$ . Although  $\mathcal{C}$  is acceptable with respect to  $\{\mathcal{C}\}$ , it is not weak-acceptable with respect to that set, as every attack requires the defense of  $\mathcal{C}$ . Self-defender arguments are relevant to elaborate a more flexible notion of acceptance. This will be addressed in Section 6.

Clearly, an argument  $\mathcal{A}$  may have a set of adjusted defenses. When  $\mathcal{A}$  is included in an admissible set  $S$ , then some or all of these adjusted defenses are included in  $S$ . Even more, the intersection of all adjusted defenses of  $\mathcal{A}$  is included in  $S$ .

**Proposition 1** Let  $D = \{S_1, S_2, \dots, S_n\}$  be the set of all adjusted defenses of an argument  $\mathcal{A}$ . For every admissible set  $T$  such that  $\mathcal{A} \in T$ , the following holds:

1.  $S_i \subseteq T$ , for some  $i$ ,  $1 \leq i \leq n$ .
2.  $\bigcap_{i=1}^{i=n} S_i \subseteq T$

*Proof:*

1. Argument  $\mathcal{A}$  is acceptable with respect to  $T$ , and therefore  $T$  is a collective defense for  $\mathcal{A}$ . Let  $V = \{X_1, X_2, \dots, X_m\}$  be the set of defeaters of  $\mathcal{A}$ . Every element of  $V$  is defeated by an argument in  $T$ . Let  $W \subseteq T$  be a minimal set of arguments such that every element in  $W$  is a defeater of an argument in  $V$ . Then

the set  $\mathcal{A}$  is acceptable with respect to  $W$ . As  $W$  is minimal, it is an adjusted defense, and therefore  $W = S_i$  for some  $i$ ,  $1 \leq i \leq n$ .

2. Trivial from previous proof.  $\square$ .

Adjusted defenses are a guarantee on acceptance. If every argument in at least one adjusted defense is accepted, then the defended argument may also be included in the acceptance set. Definition 8 can be used to compare defenses.

**Definition 13 (Forceful defense)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $S$  be a set of arguments and let  $\mathcal{A} \in \text{Args}$ . The set  $S$  is a *forceful-defense* of  $\mathcal{A}$  if  $S$  is an adjusted defense of  $\mathcal{A}$  and no other adjusted defense is a stronger defense than  $S$ .

Following the example in Figure 4, the sets  $S_1 = \{\mathcal{A}, \mathcal{D}\}$  and  $S_2 = \{\mathcal{A}, \mathcal{F}\}$  are adjusted defenses of  $\mathcal{A}$ . However,  $S_1$  is a stronger collective defense of  $\mathcal{A}$  than  $S_2$ . Therefore,  $S_1$  is a *forceful-defense* of  $\mathcal{A}$ . Note that this kind of tightened defense is not unique: the set  $S_3 = \{\mathcal{E}, \mathcal{D}\}$  is also a *forceful-defense*. Forceful defense is ideal in the sense that the strongest defenders are used, and therefore is a measure of quality for argument acceptance. An argument accepted by the use of a *forceful-defense* is strongly endorsed in the acceptance set.

**Definition 14 (Forceful argument inclusion)** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $S$  be an admissible set of arguments and let  $\mathcal{A} \in S$ . The argument  $\mathcal{A}$  is said to be *forcefully-included* in  $S$  if at least one *forceful-defense* of  $\mathcal{A}$  is included in  $S$ .

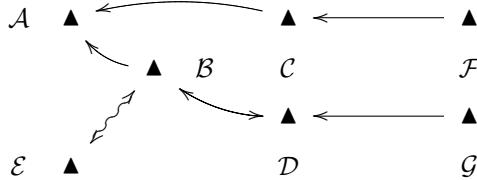


Figure 5.  $\mathcal{A}$  is not forcefully-included in  $\{\mathcal{A}, \mathcal{E}, \mathcal{F}, \mathcal{G}\}$

**Example 3** Consider the EAF of Figure 5. In the admissible set  $S = \{\mathcal{A}, \mathcal{E}, \mathcal{F}, \mathcal{G}\}$ , the argument  $\mathcal{A}$  is not *forcefully-included* in  $S$  because the adjusted defense of  $\mathcal{A}$  in that set is  $\{\mathcal{E}, \mathcal{F}\}$ , which is not the strongest collective defense. Note that  $\{\mathcal{D}, \mathcal{F}\}$  is the *forceful-defense* of  $\mathcal{A}$ , as  $\mathcal{D}$  is stronger defender than  $\mathcal{E}$ . However,  $\mathcal{D}$  is not included in  $S$  and therefore  $\mathcal{A}$  cannot be reinstated by the use of its strongest defender but  $\mathcal{E}$ .

Forceful inclusion requires the use of the strongest adjusted defenses. This resembles top-admissibility, although there is a difference in strength requirement.

**Proposition 2** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $S$  be an admissible set of arguments. If every argument in  $S$  is *forcefully-included* in  $S$ , then  $S$  is top-admissible.

*Proof:* If every argument  $\mathcal{X}$  in  $S$  is *forcefully included* in  $S$ , then every strongest defense of  $\mathcal{X}$  is included in  $S$  and therefore no other set provides a stronger defense of  $\mathcal{X}$ . Then

$S$  is top-admissible.  $\square$ .

The converse of Proposition 2, however, is not true. In Figure 5 the set  $\{\mathcal{A}, \mathcal{E}, \mathcal{F}, \mathcal{G}\}$  is top-admissible, even though argument  $\mathcal{A}$  is not forcefully-included in that set. In the following section these notions are studied in the context of Dung's grounded extension.

## 5. Grounded extension

In Dung's work, a skeptical position of argument acceptance in argumentation frameworks is captured by the grounded extension.

**Definition 15 (Grounded extension)** [9] *An admissible set  $R$  of arguments is a complete extension if and only if each argument which is acceptable with respect to  $R$ , belongs to  $R$ . A set of arguments  $G$  is a grounded extension if and only if it is the least (with respect to set inclusion) complete extension.*

The grounded extension is unique and it captures the arguments that can be directly or indirectly defended by defeater-free arguments. This extension is also the least fix-point of a simple monotonic function:

$$F_{AF}(S) = \{\mathcal{A} : \mathcal{A} \text{ is acceptable with respect to } S\}.$$

The following theorem states that this skeptical position is also based on strong defense.

**Theorem 5.1** *Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. Let  $GE$  be the grounded extension of  $\Phi$ . Then every argument in  $GE$  is forcefully-included in  $GE$ .*

*Proof:* The set  $F_\Phi(\emptyset)$  is formed by all defeater-free arguments in  $\text{Args}$ . These arguments cannot be involved in a blocking defeat as they lack of defeaters. The strongest adjusted defense for every argument in  $F_\Phi(\emptyset)$  is the empty set and thus they are forcefully included in  $F_\Phi(\emptyset)$ . Because no blocked defeat is present, any defense these arguments provide is through proper defeat, which is the strongest defense. Every argument acceptable with respect to  $F_\Phi(\emptyset)$  has a strongest adjusted defense on that set and then it is forcefully included in  $F_\Phi^2(\emptyset)$ . Suppose every argument in  $F_\Phi^k(\emptyset)$  is forcefully-included in  $F_\Phi^k(\emptyset)$ . We will prove that an argument acceptable with respect to  $F_\Phi^k(\emptyset)$  is forcefully included in  $F_\Phi^{k+1}(\emptyset)$ . Let  $\mathcal{A} \in \text{Args} - F_\Phi^k(\emptyset)$  acceptable with respect to  $F_\Phi^k(\emptyset)$  and let  $\mathcal{B}$  a defeater of  $\mathcal{A}$ . Let  $\mathcal{C} \in F_\Phi^k(\emptyset)$  be a defender of  $\mathcal{A}$  against  $\mathcal{B}$ .

- if  $\mathcal{C}$  is a proper defeater of  $\mathcal{B}$ , then  $\mathcal{C}$  is a strongest defense with respect to  $\mathcal{B}$ , favoring the forceful inclusion of  $\mathcal{A}$ .
- if  $\mathcal{C}$  is a blocking defeater of  $\mathcal{B}$ , then  $\mathcal{C}$  as part of  $F_\Phi^k(\emptyset)$  was previously defended from  $\mathcal{B}$  by an argument  $\mathcal{D} \in F_\Phi^k(\emptyset)$ . By hypothesis,  $\mathcal{C}$  is forcefully included in  $F_\Phi^k(\emptyset)$  and then  $\mathcal{D}$  is a proper defeater of  $\mathcal{B}$ . Thus,  $\mathcal{D}$  is a strongest defense with respect to  $\mathcal{B}$ , favoring the forceful inclusion of  $\mathcal{A}$ .

As in any case  $\mathcal{A}$  has a strongest defense in  $F_\Phi^k(\emptyset)$ , then  $\mathcal{A}$  is forcefully included in  $\mathcal{D} \in F_\Phi^{k+1}(\emptyset)$ .  $\square$ .

Because of Theorem 5.1, in Figure 5 it is clear that argument  $\mathcal{A}$  is not in the grounded extension, as it is not forcefully-included in any admissible set. If the argumentation framework is uncontroversial [9], then the grounded extension is the intersection of all preferred extensions. As a consequence, the skeptical acceptance with respect to preferred semantics requires dropping out every non forcefully-included argument.

## 6. Weak grounded extension

The grounded extension only captures forcefully-included arguments. It is possible to adopt a more credulous approach, expanding the acceptance by considering self-defender arguments.

**Definition 16** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. The extended characteristic function of  $\Phi$  is defined as  $F_\Phi^\cup(S) = F_\Phi(S) \cup \{\mathcal{A} : \mathcal{A} \text{ is weak acceptable with respect to } S \cup \mathcal{A}\}$

The function  $F_\Phi^\cup$  is more permissive than classic characteristic function in the sense that it allows the inclusion of self-defender arguments whenever they are partially defended by arguments in  $S$ .

**Proposition 3** If  $S$  is an admissible set of arguments, then  $F_\Phi^\cup(S)$  is admissible.

*Proof:* Clearly, if  $S$  is admissible, then  $F_\Phi(S)$  is also admissible [9]. We are going to proof that the addition of weak acceptable arguments does not disrupt the admissibility notion. Suppose  $F_\Phi^\cup(S)$  is not admissible. Then either (a) an argument is not acceptable with respect to that set, or (b) it is not conflict-free:

(a) Suppose there is an argument  $\mathcal{B} \in F_\Phi^\cup(S)$  such that  $\mathcal{B}$  is not acceptable with respect to  $F_\Phi^\cup(S)$ . As  $F_\Phi(S)$  is admissible, then clearly  $\mathcal{B}$  is weak-acceptable with respect to  $S \cup \{\mathcal{B}\}$  (it cannot belong to  $F_\Phi(S)$ ). Being a self-defender argument, some of the defeaters of  $\mathcal{B}$  are defeated by  $S$ , while the rest is defeated by  $\mathcal{B}$  itself. Thus, every defeater of  $\mathcal{B}$  has a defeater in  $S \cup \{\mathcal{B}\} \subseteq F_\Phi^\cup(S)$ . But then  $\mathcal{B}$  is acceptable with respect to  $F_\Phi^\cup(S)$ , which is a contradiction.

(b) Suppose there are two arguments  $\mathcal{A}, \mathcal{B}$  in  $F_\Phi^\cup(S)$  such that  $\{\mathcal{A}, \mathcal{B}\} \in \mathbf{C}$ . If, say,  $\mathcal{A}$  is a proper defeater of  $\mathcal{B}$ , then there exists an argument  $\mathcal{C}$  in  $S$  such that  $\mathcal{C}$  defeats  $\mathcal{A}$  ( $\mathcal{A}$  is not a dead-end defeater of  $\mathcal{B}$  and it is required to be defeated by  $S$ ). But then  $\mathcal{A}$  is not acceptable with respect to  $S$  and it can only be weak acceptable with respect to  $S \cup \{\mathcal{A}\}$ . Now  $\mathcal{C}$  cannot be a dead-end defeater of  $\mathcal{A}$  (it should not be in  $S$  then) and then  $S$  defends  $\mathcal{A}$  against  $\mathcal{C}$ , thus  $S$  is not conflict free, which is a contradiction. On the other hand, if  $\mathcal{A}$  and  $\mathcal{B}$  are blocking defeaters then at least on them, say  $\mathcal{A}$ , is a self-defender argument. Then  $\mathcal{B}$  cannot be its dead-end defeater (otherwise it is excluded) and then an argument  $\mathcal{D} \in S$  defeats  $\mathcal{B}$  ( $S$  defends  $\mathcal{A}$ ). But then  $\mathcal{B}$  is not acceptable with respect to  $S$ , and it must be also a self-defender argument. Thus, it is defended by  $S$  of every non dead-end defeater. In particular,  $\mathcal{B}$  is defended against  $\mathcal{D}$  by an argument  $\mathcal{C} \in S$ . But then  $S$  is not conflict-free, which is a contradiction.

As suppositions (a) and (b) lead to contradiction, then  $F_\Phi^\cup(S)$  is admissible.  $\square$

Following the same steps in [9], an extension can be obtained using function  $F_\Phi^\cup$ . This function is monotonic (wrt set inclusion), because an argument that is weak-acceptable with respect to  $S$  is also weak-acceptable with respect to supersets of  $S$ .

**Definition 17** Let  $\Phi = \langle \text{Args}, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$  be an EAF. The weak grounded extension is the least fixpoint of function  $F_{\Phi}^{\sqcup}(S)$ .

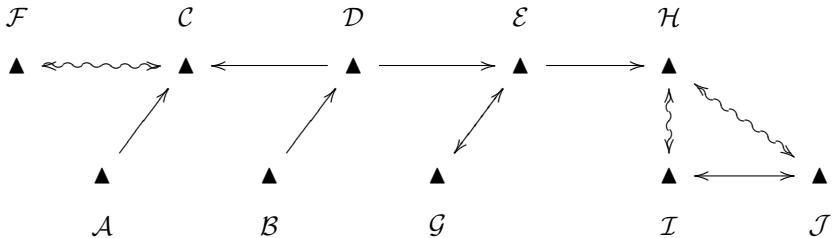


Figure 6.  $\{\mathcal{A}, \mathcal{B}, \mathcal{F}, \mathcal{E}\}$  is the weak grounded extension

**Example 4** Consider the EAF of Figure 6. The grounded extension is  $\{\mathcal{A}, \mathcal{B}, \mathcal{F}\}$ . The weak grounded extension is  $\{\mathcal{A}, \mathcal{B}, \mathcal{F}, \mathcal{E}\}$ , because argument  $\mathcal{E}$  is weak-acceptable with respect to  $\{\mathcal{B}, \mathcal{E}\}$ . Arguments  $\mathcal{I}$  and  $\mathcal{J}$  are not included in the weak grounded extension because they are not self-defenders. For example,  $\{\mathcal{H}, \mathcal{J}\}$  is an adjusted defense for  $\mathcal{I}$ . Clearly, there is a contradiction between these three arguments as the adjusted defense is not conflict free. In fact, any cycle of blocking defeaters is naturally formed by non self-defender arguments.

**Example 5** In the framework of Figure 5 the weak grounded extension is  $\{\mathcal{F}, \mathcal{G}, \mathcal{B}\}$ . In the framework of Figure 4 the grounded extension and the weak grounded extension coincides, as the only argument in a weak-acceptability condition is  $\mathcal{C}$ , but is not able to be defended from  $\mathcal{A}$  by arguments other than itself.

The addition of arguments under a weak acceptability condition is harmless in the sense of including only arguments partially defended by an admissible set. For the rest of the attacks, the argument itself is enough as defense. This allows the further inclusion of arguments that cannot be considered in the classic grounded extension.

## 7. Conclusions

In this paper we analyzed the strength of defenses in extended argumentation frameworks. In this formalism, the defeat relation is derived using a preference criterion over arguments. Thus, a *proper defeat* is consequence of preferring an argument in a conflicting pair, while a *blocking defeat* is consequence of incomparable or equivalent-in-force conflicting arguments. The defense of an argument may be achieved by proper defeaters or by blocking defeaters. Clearly, the quality of a defense depends on the type of defeaters used. Defense through proper defeaters is stronger than defense through blocking defeaters. Even more, in blocking situations, the defense provided by equivalent in force arguments may be considered stronger than the defense provided by incomparable arguments. Under this position, the force of a defense is formalized, and it is possible to evaluate how well defended an argument is when included in an admissible set. An

argument is forcefully included in an admissible set when the best defense is captured by that set. A top-admissible set is including, for every argument in the set, the strongest defense as it is possible to conform admissibility. In extended abstract frameworks, every argument in the classic grounded extension is forcefully included in that set, and then arguments with weaker defenses are dropped out. In order to adopt a slightly credulous approach, the notion of *weak acceptability* is introduced, allowing the definition of the *weak grounded extension*, where arguments can partially defend themselves. The future works will be the study of the relation between weak grounded extension and warrant extension for EAFs as defined in [12], and other semantic notions regarding blocking defeaters, as *x-admissibility*, which was previous presented in [13].

## 8. Acknowledgments

The authors would like to thank anonymous referees for comments that led to substantial improvements and error corrections regarding weak acceptability.

## References

- [1] Leila Amgoud and Claudette Cayrol. On the acceptability of arguments in preference-based argumentation. In *14th Conference on Uncertainty in Artificial Intelligence (UAI'98)*, pages 1–7. Morgan Kaufmann, 1998.
- [2] Leila Amgoud and Claudette Cayrol. A reasoning model based on the production of acceptable arguments. In *Annals of Mathematics and Artificial Intelligence*, volume 34, 1-3, pages 197–215. 2002.
- [3] Leila Amgoud and Laurent Perrussel. Arguments and Contextual Preferences. In *Computational Dialectics-Ecai workshop (CD2000)*, Berlin, August 2000.
- [4] Pietro Baroni and Massimiliano Giacomin. Evaluation and comparison criteria for extension-based argumentation semantics. In *Proc. of I International Conf. on Computational Models of Arguments, COMMA 2006*, pages 157–168, 2006.
- [5] T.J.M. Bench-Capon. Value-based argumentation frameworks. In *Proc. of Nonmonotonic Reasoning*, pages 444–453, 2002.
- [6] Martin Caminada. Semi-stable semantics. In *Proceedings of I International Conference on Computational Models of Arguments, COMMA 2006*, pages 121–130, 2006.
- [7] Claudette Cayrol, Sylvie Doutre, Marie-Christine Lagasque-Schiex, and Jérôme Mengin. “minimal defence”: a refinement of the preferred semantics for argumentation frameworks. In *NMR*, pages 408–415, 2002.
- [8] D.C. Coste-Marquis, C. Devred, and P. Marquis. Prudent semantics for argumentation frameworks. In *17th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2005.*, 2005.
- [9] Phan M. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning and Logic Programming. In *Proc. of the 13th. IJCAI 93.*, pages 852–857, 1993.
- [10] D.C. Martínez, A.J. García, and G.R. Simari. On acceptability in abstract argumentation frameworks with an extended defeat relation. In *Proc. of I Intl. Conf. on Computational Models of Arguments, COMMA 2006*, pages 273–278, 2006.
- [11] D.C. Martínez, A.J. García, and G.R. Simari. Progressive defeat paths in abstract argumentation frameworks. In *Proceedings of the 19th Conf. of the Canadian Society for Computational Studies of Intelligence 2006*, pages 242–253, 2006.
- [12] D.C. Martínez, A.J. García, and G.R. Simari. Modelling well-structured argumentation lines. In *Proc. of XX IJCAI-2007.*, pages 465–470, 2007.
- [13] D.C. Martínez, A.J. García, and G.R. Simari. On defense strength of blocking defeaters in admissible sets. In *To appear in Proceedings of Second International Conference on Knowledge Science, Engineering and Management KSEM 07. Melbourne, Australia.*, 2007.
- [14] Gerard A. W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90(1–2):225–279, 1997.

# Basic influence diagrams and the liberal stable semantics

Paul-Amaury MATT and Francesca TONI

Imperial College London, Department of Computing

**Abstract.** This paper is concerned with the general problem of constructing decision tables and more specifically, with the identification of all possible outcomes of decisions. We introduce and propose *basic influence diagrams* as a simple way of describing problems of decision making under strict uncertainty. We then establish a correspondence between basic influence diagrams and symmetric generalised assumption-based argumentation frameworks and adopt an argumentation-based approach to identify the possible outcomes. We show that the intended solutions are best characterised using a new semantics that we call *liberal stability*. We finally present a number of theoretical results concerning the relationships between liberal stability and existing semantics for argumentation.

**Keywords.** Abstract and assumption-based argumentation, decision tables and decision making.

## Introduction

In decision theory [French, 1987], a decision making problem is modelled in terms of a decision table representing the outcomes of decisions under different scenarios, a probability distribution representing the likelihood of occurrence of the scenarios and a utility function measuring the subjective value of each possible outcome. In general, the construction of a decision table is not straightforward, for several reasons. Firstly, one must choose a way of representing the decisions, scenarios and possible outcomes. Secondly, finding an exhaustive set of disjoint scenarios and being able to describe the outcome of each decision under each one of them is a long process which requires expert knowledge in the practical domain of application. Thirdly, identifying the possible outcomes of decisions implicitly involves reasoning in the presence of conflicts that capture what is materially impossible or epistemically inconceivable. This paper presents a paradigm based on logical argumentation for finding all possible outcomes of decisions and constructing decision tables.

This approach is justified insofar as logical argumentation allows to represent decisions, scenarios and outcomes using literals, to capture their logical dependencies, to reason about them and resolve conflicts in a rational way. Concretely, we adopt *assumption-based argumentation* [Bondarenko et al., 1997; Dung et al., 2006; Toni, 2007] for identifying the possible outcomes of decisions, given *basic influence diagrams*, a simple tool for describing problems of decision making under strict uncertainty. Diagrams which are very similar in essence have already been deployed to feed into (a different form of) argumentation in [Morge and Mancarella, 2007].

The paper is organised as follows. Section 1 introduces basic influence diagrams. Section 2 recalls background definitions for assumption-based argumentation and introduces a generalisation needed for the purposes of this paper. Section 3 shows how to transform basic influence diagrams into argumentation frameworks. Then section 4 presents and justifies the conditions under which the solutions of basic influence diagrams are "rational". Section 5 shows that existing semantics of argumentation for conflict resolution mismatch with these conditions and proposes the *liberal stable semantics* as an alternative. The properties of liberal stability and its relationships with other semantics are then studied in detail. Section 6 summarises and concludes the paper.

## 1. Basic influence diagrams

The problem we study can be summarised by the following question:

*What are all the possible outcomes of our decisions in a given decision domain ?*

In order to construct a decision table, it is necessary to identify all possible outcomes and not just simply the best or most likely ones. We introduce *basic influence diagrams* to model the decision maker's goals, decisions, uncertainties and beliefs, their causal dependencies and possible conflicts. These diagrams are very similar in essence to Bayesian (or belief) networks [Pearl, 1986] and influence diagrams [Howard and Matheson, 1981] but are simpler, non-numerical data structures. Such diagrams are widely used for knowledge representation in artificial intelligence and recently, simpler qualitative forms of diagrams have started to be used to structure argumentation-based systems for decision support [Morge and Mancarella, 2007]. Formally, a *basic influence diagram* is composed of two parts: a diagram and a set of dependency rules. The diagram is an annotated finite directed acyclic graph whose nodes are literals of the form  $p$  or  $\neg p$  where  $p$  is a proposition in some given language  $\mathcal{L}$ . Every literal belongs to one of the following categories:

- *Goals* are the nodes that have no outgoing arcs. Goals represent what the decision maker ultimately desires to achieve (positive goals) or avoid (negative goals). Positive/negative goals are graphically distinguished with a +/- symbol.
- *Atomic decisions* are an arbitrary strict subset of the nodes that are not goals and that have no incoming arcs. Atomic decisions are graphically distinguished from the other nodes by a squared box.
- *Beliefs* are the nodes which are neither goals nor decisions. Beliefs do not have any particular distinguishing feature. The beliefs that have no incoming arcs are called *fundamental beliefs*. The fundamental beliefs that are known to be true are called *facts* and are underlined. The other fundamental beliefs are called *unobservable fundamental beliefs* and are annotated with a ? symbol.

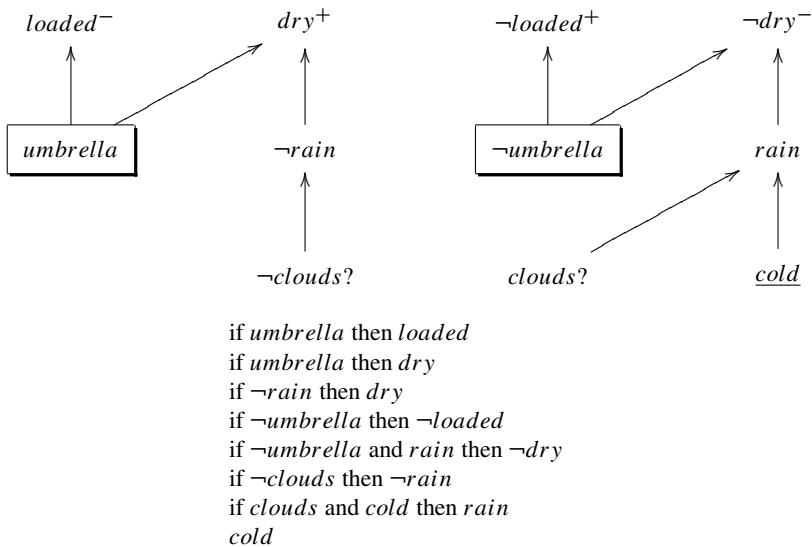
An arc from node  $p$  to node  $q$  means that the truth of the literal  $q$  logically depends on the truth of the literal  $p$ . Logical dependencies are expressed by *dependency rules*, which may be of the form

- "if  $p_1$  and ... and  $p_k$  then  $q$ ", or
- " $q$ " as fact or observation

where  $p_1, \dots, p_k$  and  $q$  are literals. The arcs of the influence diagram and the dependency rules must conform in the following way. For each rule of the form "if  $p_1$  and ...

and  $p_k$  then  $q$ " there must exist an arc from  $p_i$  to  $q$  for every  $i \in \{1, \dots, k\}$ . For each dependency rule " $q$ ",  $q$  must appear as a fact in the diagram. Conversely, for each arc from  $p$  to  $q$  in the diagram, there must be at least one dependency rule "if ...  $p$  ... then  $q$ ", and for each fact  $q$  there must exist a dependency rule " $q$ ".

Let us now see how to concretely represent a decision making problem using a basic influence diagram. The basic influence diagram<sup>1</sup> shown in figure 1 models a simple decision problem borrowed and adapted from [Amgoud and Prade, 2004], in which one has to decide whether to take an umbrella or not when going out for a walk. The decision maker has some elementary knowledge about the weather. He knows that the weather is cold, and he knows that if it is cold and there are clouds, then it must be raining. Taking an umbrella guarantees him to be able to stay dry but has the disadvantage of making him loaded.



**Figure 1.** Basic influence diagram corresponding to the umbrella example.

Dependency rules are purely deterministic, unlike probabilistic dependencies in Bayesian networks. In basic influence diagrams, uncertainties are not quantified, as probabilities play no role in the identification of the possible outcomes. Similarly, basic influence diagrams do not have a utility/value node as in standard influence diagrams. Decision tables should display all possible outcomes and not just those with maximal utility. Moreover, a utility function may only be defined once the set of outcomes has been identified. Although diagrams are not essential for representing knowledge in decision making under strict uncertainty they are nevertheless an excellent graphical aid to the specification of dependency rules.

Our final remark on basic influence diagrams concerns the use of negation  $\neg$ . First, when a literal  $p$  is a node of the diagram, its negation  $\neg p$  is not required to be in the diagram. In the example,  $\neg cold$  is not a literal of interest. Second, whenever a literal in

<sup>1</sup>In this example, the literal/node *dry* is true when either of its predecessors *umbrella* or  $\neg$ *rain* is true. On the opposite, the literal/node  $\neg$ *dry* is true when both of its predecessors  $\neg$ *umbrella* and *rain* are true. This is not implied by the diagram and justifies the use of dependency rules.

the diagram admits a negation, one must make sure that practically, in the decision domain, either  $p$  holds or  $\neg p$  holds. Third, cases where both  $p$  and  $\neg p$  hold are considered as flawed or invalid opinions about the situation. Finally, cases where neither  $p$  nor  $\neg p$  hold are considered as incomplete opinions about the situation. Therefore, the user of basic influence diagrams shall be aware that strong logical dependencies exist between nodes of the form  $p$  and  $\neg p$ .

## 2. Generalised assumption-based argumentation

This section provides background on assumption-based argumentation [Bondarenko *et al.*, 1997; Dung *et al.*, 2006; Toni, 2007; Dung *et al.*, 2007] and introduces a simple generalisation of this form of argumentation. An *assumption-based argumentation framework* is a tuple  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$  where

- $(\mathcal{L}, \mathcal{R})$  is a *deductive system*, consisting of a language  $\mathcal{L}$  and a set  $\mathcal{R}$  of inference rules,
- $\mathcal{A} \subseteq \mathcal{L}$ , referred to as the set of *assumptions*,
- $\mathcal{C}: \mathcal{A} \mapsto \mathcal{L}$  is a total mapping associating a *contrary* sentence to each assumption.

The elements of  $\mathcal{L}$  are called sentences. The inference rules in  $\mathcal{R}$  have the syntax  $\frac{p_1, \dots, p_n}{q}$  (for  $n \geq 0$ ) where  $p_i \in \mathcal{L}$ . We will refer to  $q$  and  $p_1, \dots, p_n$  as the *conclusion* and the *premises* of the rule, respectively. We restrict our attention to *flat* frameworks, such that if  $q \in \mathcal{A}$ , then there exists no inference rule of the form  $\frac{p_1, \dots, p_n}{q} \in \mathcal{R}$ , for any  $n \geq 0$ .

A *generalised assumption-based argumentation framework* is an assumption-based argumentation framework with a *generalised notion of contrary*, as follows:  $\mathcal{C} \subseteq 2^{\mathcal{A}} \times \mathcal{L}$ . For every  $(A, p) \in \mathcal{C}$ ,  $A \cup \{p\}$  represents a set of sentences that cannot hold together, with  $A$  the designated "culprits" to be withdrawn should such a combination arise. Contraries as in standard assumption-based argumentation can be represented in terms of this generalised notion of contrary, by having a pair for each assumption, consisting of a singleton culprit set with that assumption and its contrary sentence. In the remainder, when clear from the context, we will refer to a generalised assumption-based argumentation framework simply as assumption-based argumentation framework.

In the assumption-based approach to argumentation, arguments are (forward or backward) deductions to conclusions, based upon assumptions.

**Definition 1 (forward argument)** An argument  $A \vdash_f p$  with conclusion  $p$  based on a set of assumptions  $A$  is a sequence  $\beta_1, \dots, \beta_m$  of sentences in  $\mathcal{L}$ , where  $m > 0$  and  $p = \beta_m$ , such that, for all  $i = 1, \dots, m$ :  $\beta_i \in A$ , or there exists  $\frac{a_1, \dots, a_n}{\beta_i} \in \mathcal{R}$  such that  $a_1, \dots, a_n \in \{\beta_1, \dots, \beta_{i-1}\}$ .

**Definition 2 (tight or backward argument)** Given a selection function, a tight argument  $A \vdash_b p$  with conclusion  $p$  based on a set of assumptions  $A$  is a sequence of multi-sets  $S_1, \dots, S_m$ , where  $S_1 = \{p\}$ ,  $S_m = A$ , and for every  $1 \leq i \leq m$ , where  $\sigma$  is the selected sentence occurrence in  $S_i$ : 1) If  $\sigma$  is a non-assumption sentence then  $S_{i+1} = S_i - \{\sigma\} \cup S$  for some inference rule of the form  $\frac{S}{\sigma} \in \mathcal{R}$ . 2) If  $\sigma$  is an assumption then  $S_{i+1} = S_i$ .

Basically, tight arguments restrict the set  $A$  to include only assumptions that are relevant to the argument conclusion. Forward and backward arguments with the same conclusion are linked by

**Theorem 1 ([Dung et al., 2006])** *For every tight argument with conclusion  $p$  supported by  $A$  there exists an argument with conclusion  $p$  supported by  $A$ . For every argument with conclusion  $p$  supported by  $A$  and for every selection function, there exists a tight argument with conclusion  $p$  supported by some subset  $A' \subseteq A$ .*

In order to determine whether a conclusion (set of sentences) should be drawn, a set of assumptions needs to be identified providing an “acceptable” support for the conclusion. Various notions of “acceptable” support can be formalised [Dung, 1995; Bondarenko et al., 1997; Dung et al., 2002; 2006; 2007; Toni, 2007; Caminada, 2006], using a notion of attack amongst sets of assumptions. In this paper, for any sets of assumptions  $A$  and  $B$  we say that

**Definition 3 (attack)**  *$A$  attacks  $B$  if and only if there exists a pair  $(P, q) \in \mathcal{C}$  such that  $A \vdash_f q$  and  $P \subseteq B$ .*

We may transpose existing semantics for abstract [Dung, 1995] and assumption-based argumentation [Dung et al., 2006] to the case of generalised assumption-based argumentation by saying that a set of assumptions  $A$  is

- *conflict-free* iff  $A$  does not attack itself
- *naive* iff  $A$  is maximally conflict-free
- *admissible* iff  $A$  is conflict-free and  $A$  attacks every set of assumptions  $B$  that attacks  $A$
- *stable* iff  $A$  is conflict-free and attacks every set it does not include
- *semi-stable* iff  $A$  is complete where  $\{A\} \cup \{B | A \text{ attacks } B\}$  is maximal
- *preferred* iff  $A$  is maximally admissible
- *complete* iff  $A$  is admissible and includes every set  $B$  such that  $A$  attacks all sets attacking  $B$  ( $A$  defends  $B$ )
- *grounded* iff  $A$  is minimally complete
- *ideal* iff  $A$  is admissible and included in every preferred set.

The definitions given here are exactly the same as those used for standard assumption-based argumentation except for the stable and complete semantics which cannot be directly applied in generalised assumption-based argumentation and need a slight generalisation, and for the semi-stable semantics which has only been defined in the context of abstract argumentation [Caminada, 2006]. The new definitions for the stable and complete semantics collapse to the standard ones in every instance of a generalised framework where culprit sets of assumptions are singletons, as in standard assumption-based argumentation (cf. proof in appendix). The new definition for the semi-stable semantics is a possible adaptation of the original one to the case of generalised frameworks.

### 3. Transforming basic influence diagrams into argumentation frameworks

Basic influence diagrams can best be analysed using generalised assumption-based argumentation. Given a basic influence diagram, we define

- $\mathcal{L}$  as the set of all literals in the influence diagram
- $\mathcal{R}$  as the set of all inference rules of the form
  - \*  $\frac{p_1, \dots, p_n}{q}$  where "if  $p_1$  and ... and  $p_n$  then  $q$ " is a dependency rule, or
  - \*  $\frac{}{q}$  where " $q$ " is a dependency rule
- $\mathcal{A}$  as the set of all atomic decisions and unobservable fundamental beliefs
- $\mathcal{C}$  as the set of all pairs  $(P, q)$  such that  $(q, \neg q) \in \mathcal{L}^2$  and  $P \vdash_b \neg q$ .

Here,  $\mathcal{L}$  represents the set of nodes of the diagram and  $\mathcal{R}$  its arcs and dependency rules.  $\mathcal{A}$  represents the uncertainties of the decision maker: he does not know which (atomic) decisions to make and which unobservable fundamental beliefs are true.  $\mathcal{C}$  implicitly captures the conflicts between literals  $q$  and the relevant assumptions  $P$  supporting their negation  $\neg q$ . The notion of attack used in this paper is such that  $A$  attacks  $B$  if and only if there exists  $(p, \neg p) \in \mathcal{L}^2$  such that  $A \vdash_f p$  and  $B \vdash_f \neg p$  (this can be easily proved using theorem 1). The assumption-based framework constructed captures in fact all the information originally contained in the basic influence diagram, except the positivity (+) and negativity (-) of goals which are not important for identifying all possible outcomes.

Since atomic decisions and unobservable fundamental beliefs have no incoming nodes, they are not conclusions of any inference rule. Therefore, the frameworks constructed are always guaranteed to be flat. Besides, these frameworks have the property of symmetry, which play a substantial role in the proofs given in this paper.

**Property 1 (symmetry)** *If  $A$  attacks  $B$ , then  $B$  attacks  $A$ .*

**Proof 1** *If  $A$  attacks  $B$ , then there exists  $(P, q) \in \mathcal{C}$  such that  $P \subseteq B$  and  $A \vdash_f q$ . By definition of the contrary relation,  $P \vdash_b \neg q$  is a tight argument. By definition,  $\mathcal{C}$  also contains all pairs of the form  $(Q, \neg q)$  where  $Q \vdash_b q$  is a tight argument. Since  $A \vdash_f q$  is an argument, there exist by theorem 1 a tight argument of the form  $Q \vdash_b q$  such that  $Q \subseteq A$ .  $P \vdash_b \neg q$  and  $P \subseteq B$  so by theorem 1 one also has  $B \vdash_f \neg q$ . In summary,  $(Q, \neg q) \in \mathcal{C}$ ,  $Q \subseteq A$  and  $B \vdash_f \neg q$ , so  $B$  attacks  $A$ .*

In the remainder of the paper we show that the problem of finding the "rational" possible outcomes of decisions with respect to a given basic influence diagram may be reduced to the problem of finding "liberal stable" sets of assumptions in the corresponding argumentation framework.

#### 4. Rationality for basic influence diagrams

When reasoning with rules, rationality is based on two requirements: consistency, or absence of conflicts, and closure under inference [Caminada and Amgoud, 2007; Toni, 2007]. These two properties are widely recognised as essential and we believe that a rational decision maker should conform to them. When using basic influence diagrams, it is however important to strengthen this basic notion of rationality by adding a third property. Basic influence diagrams allow the decision maker to express conflicts between beliefs in the form of contrary literals, such as *dry* and  $\neg$ *dry* or *rain* and  $\neg$ *rain*. Here,  $\neg$  is meant to be used as classical negation, i.e. if  $p$  holds, then its contrary  $\neg p$  does not hold, but if  $p$  does not hold, then its contrary  $\neg p$  holds. So, for every pair  $(p, \neg p) \in \mathcal{L}^2$ , one must enforce the decision maker to choose to believe in either  $p$  or  $\neg p$ . We call

this property *decidedness*. Some authors [Takahashi and Sawamura, 2004] insist on the philosophical importance of allowing descriptions of states of the world whereby both  $p$  and  $\neg p$  hold, or even where neither of them holds. Such descriptions correspond for instance to logical formalisations of dialogues between parties sharing different views, but they are not meant to be taken as fully rational opinions of individuals. Therefore, we adopt the following

**Definition 4 (rational outcome)** An outcome  $O \subseteq \mathcal{L}$  is rational if and only if

- $\forall(p, \neg p) \in \mathcal{L}^2$ , it is not the case that  $p \in O$  and  $\neg p \in O$  (consistency)
- $\forall(p, \neg p) \in \mathcal{L}^2$ ,  $p \in O$  or  $\neg p \in O$  (decidedness)
- there exists a set of assumptions  $A$  such that  $O = \{p \in \mathcal{L} \mid A \vdash_f p\}$  (closure under dependency rules)

Rational outcomes may contain positive and/or negative goals. When analysing a decision problem, it is indeed important to identify all possible outcomes, whether these are good or bad for the decision maker. Of course, the decision maker will try to achieve only the best ones, but ignoring bad possible outcomes would be quite imprudent.

Concerning the previous definition, note also that the closure property is expressed in terms of the assumption-based framework on which dependency rules and influence diagrams are mapped. Indeed, the notion  $\vdash_f$  there corresponds exactly to the notion of reasoning with dependency rules we are after. Finally, it can be remarked that the set  $A$  is always unique and given by  $A = O \cap \mathcal{A}$ . In this paper, rational opinions of the decision maker correspond to rational outcomes.

**Definition 5 (rational opinion)** A rational opinion is a set of assumptions  $A$  such that the set  $O(A)$  defined as  $\{p \in \mathcal{L} \mid A \vdash_f p\}$  is a rational outcome.

Identifying all rational outcomes is equivalent to identifying all rational opinions. In the next section, we introduce a new argumentation semantics that allows to characterise rational opinions.

## 5. Liberal stable semantics

In general, the conflict-freeness of  $A$  is sufficient to guarantee the consistency of  $O(A)$ .

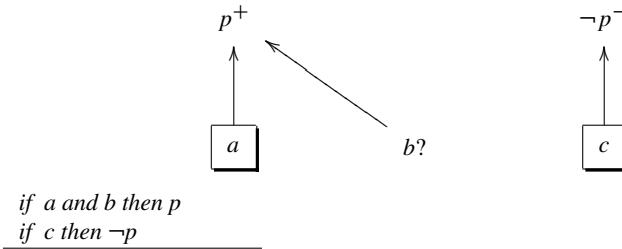
**Lemma 1 (consistency)**  $\exists p \in \mathcal{L}$  such that  $p \in O(A)$  and  $\neg p \in O(A)$  iff  $A$  does not attack itself.

**Proof 2** It is equivalent to prove that  $\exists p \in \mathcal{L}$  such that  $p \in O(A)$  and  $\neg p \in O(A)$  iff  $A$  attacks itself. By definition of an attack,  $A$  attacks itself iff  $\exists(P, q) \in \mathcal{C}$  such that  $P \subseteq A$  and  $A \vdash_f q$ . By definition of the contrary relation,  $A$  attacks itself iff there exists a tight argument  $P \vdash_b \neg q$  such that  $P \subseteq A$  and  $A \vdash_f q$ . By theorem 1, this is also equivalent to saying that  $A$  attacks itself iff there exists  $q \in \mathcal{L}$  such that  $A \vdash_f \neg q$  and  $A \vdash_f q$ .

Unfortunately, the semantics mentioned in the previous section fail to characterise the rationality of opinions.

**Theorem 2** None of the notions of acceptability in section 2 are such that  $O(A)$  is a rational outcome if and only if  $A$  is acceptable.

**Proof 3** Let us consider the following basic influence diagram and influence rules



We obtain a generalised assumption-based framework with  $\mathcal{L} = \{a, b, c, p, \neg p\}$ ,  $\mathcal{R} = \{\frac{a,b}{p}, \frac{c}{\neg p}\}$ ,  $\mathcal{A} = \{a, b, c\}$  and  $\mathcal{C} = \{(\{a, b\}, \neg p), (\{c\}, p)\}$ . The rational opinions are  $\{c\}$ ,  $\{a, b\}$ ,  $\{a, c\}$  and  $\{b, c\}$ .  $\{\}$  is conflict-free but is not rational.  $\{c\}$  and  $\{a, c\}$  cannot be both naive but are nevertheless both rational.  $\{\}$  is admissible but is not rational.  $\{c\}$  is not stable but is nevertheless rational.  $\{c\}$  and  $\{a, c\}$  cannot both be preferred but are nevertheless both rational.  $\{c\}$  and  $\{a, c\}$  cannot both be grounded but are nevertheless both rational.  $\{\}$  is ideal but is not rational.  $\{c\}$  is not complete (it defends  $\{a\}$  (since  $\{a\}$  is not attacked) and does not contain  $a$ ) but is nevertheless rational.  $\{c\}$  is not complete and a fortiori not semi-stable, but is nevertheless rational.

This justifies the need for a new semantics. For any  $A \subseteq \mathcal{A}$ , let us denote  $att(A) = \{B \subseteq \mathcal{A} \mid A \text{ attacks } B\}$ . We define

**Definition 6 (liberal stability)**  $A$  is liberal stable if and only if

- $A$  is conflict-free, and
- there is no conflict-free set  $B$  such that  $att(B) \supset att(A)$ .

For an abstract argumentation framework [Dung, 1995] with set of arguments  $Arg$  and attack relation  $R \subseteq Arg \times Arg$ , liberal stability can be defined as follows. If we denote  $att(S) = \{B \in Arg \mid \exists A \in S, A R B\}$  for any  $S \subseteq Arg$ , then the set of arguments  $S \subseteq Arg$  is liberal stable if and only if  $S$  is conflict-free and there is no conflict-free set  $X \subseteq Arg$  such that  $att(X) \supset att(A)$ .

This new semantics enables to exactly identify the rational opinions in the example of proof 3. Indeed, note first that the conflict-free sets of assumptions are the strict subsets of  $\mathcal{A}$ . It is clear that if a set of assumptions  $A$  is such that both  $A \not\models_f p$  and  $A \not\models_f \neg p$  then  $att(A) = \emptyset$ . Such a set is not liberal stable as for instance with the conflict-free set  $B = \{c\}$ , we have  $att(B) = \{\{a, b\}, \{a, b, c\}\} \supset \emptyset$ . So, it is necessary for a liberal stable set to be such that  $A \vdash_f p$  or  $A \vdash_f \neg p$ . The only conflict-free set such that  $A \vdash_f p$  is  $\{a, b\}$  and the only conflict-free sets such that  $A \vdash_f \neg p$  are  $\{c\}$ ,  $\{a, c\}$  and  $\{b, c\}$ . For all these sets, the only  $B$  such that  $att(B) \supset att(A)$  is  $B = \mathcal{A}$  which is not conflict-free. This proves that these sets are the liberal stable ones and as we have seen before they exactly correspond to the rational opinions. We prove more generally that this holds whenever the argumentation framework has the property of

**Hypothesis 1 (extensibility)** Every conflict-free set is contained in at least one rational opinion.

Extensibility holds in the previous example: the conflict-free sets are the strict subsets of  $\mathcal{A}$ , so either they do not contain  $a$  and are included in the rational set  $\{b, c\}$ , or they do

not contain  $b$  and are included in the rational set  $\{a, c\}$ , or they do not contain  $c$  and are included in the rational set  $\{a, b\}$ . The main result of this paper is the following

**Theorem 3** *i) If  $A$  is rational then  $A$  is liberal stable. Under extensibility, it also holds that ii) if  $A$  is liberal stable then  $A$  is rational.*

**Proof 4** According to the consistency lemma 1 and the definition of rational outcome, if  $A$  is rational then  $O(A)$  is consistent and  $A$  is conflict-free. Conversely, if  $A$  is liberal stable then  $A$  is conflict-free and  $O(A)$  is consistent. Consequently, in the remainder of the proof, in i) it only remains to prove that there is no conflict-free set attacking a strict superset of  $att(A)$  and in ii) that for all  $(p, \neg p) \in \mathcal{L}^2$ ,  $p \in O(A)$  or  $\neg p \in O(A)$ .

*Proof of i):* Let  $A$  be a rational set. Assume that there is a conflict-free set  $B$  such that  $att(B) \supset att(A)$ . Then  $att(B) - att(A) \neq \emptyset$ , i.e. there exists  $C \in att(B) - att(A)$ . So, there exist  $(p, \neg p) \in \mathcal{L}^2$  and a tight argument of the form  $P \vdash_b \neg p$  such that:  $(P, p) \in \mathcal{C}$ ,  $P \subseteq C$  and  $B \vdash_f p$ . Since  $A$  does not attack  $C$ , we also have  $A \nvdash_f p$ .  $A$  is rational and therefore  $A \vdash_f \neg p$  or  $A \vdash_f p$ . We know that  $A \nvdash_f p$  so  $A \vdash_f \neg p$ . Since  $B \vdash_f p$  and by theorem 1, we can find a tight argument  $Q \vdash_b p$  where  $Q \subseteq B$ . By definition of  $\mathcal{C}$ :  $(Q, \neg p) \in \mathcal{C}$ . In summary, we have  $(Q, \neg p) \in \mathcal{C}$ ,  $Q \subseteq B$  and  $A \vdash_f \neg p$  so  $A$  attacks  $B$ .  $B \in att(A)$  and  $att(B) \supset att(A)$  implies  $B \in att(B)$ , which is absurd because  $B$  is conflict-free.

*Proof of ii):* Let  $A$  be a liberal stable set and  $(p, \neg p) \in \mathcal{L}^2$ . Assume that  $p \notin O(A)$  and  $\neg p \notin O(A)$ .  $A$  is liberal stable and a fortiori conflict-free. Since we have assumed that the framework is extensible,  $A$  can be extended to a set  $A' \supseteq A$  that is rational.  $A'$  is therefore conflict-free and notably such that  $A' \vdash_f p$  or  $A' \vdash_f \neg p$ . We may assume without loss of generality that  $A' \vdash_f p$ . In an assumption-based framework drawn from a basic influence diagram, every sentence of  $\mathcal{L}$  is supported by at least one (tight) argument. Let then  $Q \vdash_b \neg p$  be a tight argument supporting  $\neg p$ . Clearly,  $A' \supseteq A$  implies that  $att(A') \supseteq att(A)$ . Note that in fact  $A'$  is conflict-free and  $att(A') \supset att(A)$  (since  $A'$  attacks  $Q$  but  $A$  does not attack it). This is absurd because  $A$  is liberal stable.

Fortunately, extensibility holds whenever naive opinions are decided. Extensibility is therefore a quite natural property of frameworks derived from basic influence diagrams.

**Property 2** *i)  $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$  is extensible if and only if ii) for every naive set  $N$  of assumptions, the opinion  $O(N)$  is decided.*

**Proof 5** *i)  $\Rightarrow$  ii)* Let  $N$  be a naive set. Assume  $O(N)$  is not decided. Then by conflict-freeness of  $N$  and i),  $N$  can be extended to a larger set  $N' \supseteq N$  that is rational.  $N'$  is rational and a fortiori consistent, i.e conflict-free by the consistency lemma 1.  $O(N')$  is decided, and therefore it is clear that  $N' \supset N$ .  $N'$  is a strict superset of  $N$  and is conflict-free, which contradicts the naiveness of  $N$ .

*ii)  $\Rightarrow$  i)* Let  $A$  be a conflict-free set. Then  $A$  is contained in a maximally conflict-free (naive) set  $N$ .  $N$  is rational since  $O(N)$  satisfies the property of decidedness (by ii) and the property of consistency by the consistency lemma 1.

In the umbrella example, the naive sets and their corresponding opinions are

- $N_1 = \{\text{umbrella, clouds}\}$ ,  $O(N_1) = \{\text{umbrella, clouds, loaded, dry}\}$ ,
- $N_2 = \{\text{umbrella, } \neg\text{clouds}\}$ ,  $O(N_2) = \{\text{umbrella, } \neg\text{clouds, loaded, dry}\}$ ,
- $N_3 = \{\neg\text{umbrella, } \neg\text{clouds}\}$ ,  $O(N_3) = \{\neg\text{umbrella, } \neg\text{clouds, } \neg\text{loaded, dry}\}$ ,
- $N_4 = \{\neg\text{umbrella, clouds}\}$ ,  $O(N_4) = \{\neg\text{umbrella, clouds, } \neg\text{loaded, } \neg\text{dry}\}$ .

The four opinions are decidedness and the framework is therefore extensible. In fact, the naive sets here correspond exactly to the liberal stable sets. There are therefore four possible outcomes and these solutions are those we would intuitively expect.

Frameworks derived from basic influence diagrams may not be extensible. For instance, a framework is not extensible when it admits at least one conflict-free set but does not admit any rational set/possible outcome. This happens e.g. with  $\mathcal{L} = \{d, p, \neg p\}$   $\mathcal{R} = \{\frac{d}{p}, \frac{p}{\neg p}\}$ ,  $\mathcal{A} = \{d\}$ ,  $\mathcal{C} = \{(\{d\}, p), (\{d\}, \neg p)\}$ . Such frameworks/diagrams are "pathological" and do not conform to the idea that in the practical domain of decision, either  $p$  or  $\neg p$  holds. Frameworks drawn from diagrams are not extensible either when they do not admit any conflict-free sets of assumptions. This occurs when it is possible to infer both  $p$  and  $\neg p$  simply from facts of the diagram. Such frameworks are inconsistent and all semantics including the liberal stable semantics are then anyway empty. Extensibility is a stronger notion than consistency, i.e. whenever a framework is extensible, it is also consistent. In general, one has

**Property 3** *The existence of conflict-free sets implies the existence of liberal stable ones.*

Note that there may not always exist a conflict-free sets of assumption. For instance, if in the basic influence diagram  $q$  and  $\neg q$  are facts, then in the corresponding framework, {} attacks itself, and as a matter of consequence, all sets of assumptions also attack themselves. The following theorem links the semantics of liberal stability to the others.

**Theorem 4** *Every stable set is liberal stable and every liberal stable set is conflict-free and admissible. If extensibility holds, then every naive, stable or preferred set is liberal stable and every liberal stable set is conflict-free and admissible (cf. figure 2).*

Semantics $s$	$s \Rightarrow$ liberal stable ?	liberal stable $\Rightarrow s$ ?
conflict-free	$\times^*$	$\checkmark^0$
naive	$E^1$	$\times^*$
admissible	$\times^*$	$\checkmark^2$
stable	$\checkmark^3$	$\times^*$
semi-stable	$\times^4$	$\times^*$
preferred	$E^5$	$\times^*$
complete	$\times^6$	$\times^*$
grounded	$\times^7$	$\times^*$
ideal	$\times^*$	$\times^8$

**Figure 2.** Links between liberal stability and existing semantics for argumentation.  $\checkmark$  denotes an implication that holds in general,  $E$  an implication that holds under extensibility and  $\times$  an implication that does not generally hold, even under the extensibility hypothesis. The symbols in exponent indicate the paragraph in which the result is proved.

**Proof 6** All the results marked with \* directly follow from the proof of theorem 2. We now denote  $\text{cont}(A) = \{p \in \mathcal{L} | A \vdash_f p \text{ and } (p, \neg p) \in \mathcal{L}^2\}$ .

- 0) Every liberal stable set is conflict-free by definition.
- 1) Let  $A$  be a naive set.  $A$  is conflict-free. Assume there exists a conflict-free set  $B$  such that  $\text{att}(B) \supset \text{att}(A)$ . Then, we would have  $\text{cont}(B) \supset \text{cont}(A)$ . There would then exist  $p \in$

$\text{cont}(B) - \text{cont}(A)$ . A  $\vdash_f \neg p$  otherwise A could be extended by extensibility to a conflict-free set that entails  $p$  or  $\neg p$  and that set would be strictly larger than A. This would contradict the maximality of A. So,  $\neg p \in \text{cont}(A)$ . Since  $\text{cont}(B) \supset \text{cont}(A)$ ,  $\neg p \in \text{cont}(B)$ . Since  $B \vdash_f p$ , there exists by theorem 1 a tight argument  $P \vdash_b p$  with  $P \subseteq B$ . We have  $(P, \neg p) \in \mathcal{C}$ ,  $P \subseteq B$  and  $B \vdash_f \neg p$ : B attacks itself, which is absurd.

2) In symmetric abstract frameworks, admissibility collapses with conflict-freeness [Coste-Marquis et al., 2005]. The same thing happens in symmetric frameworks derived from basic influence diagrams. Since liberal stability implies conflict-freeness, it is also clear that liberal stability implies admissibility.

3) Let A be a stable set. A attacks all the sets that it is possible to attack without attacking itself, so A is definitely liberal stable.

5) Preferred sets are defined as maximally admissible and by symmetry actually coincide with the maximally conflict-free (i.e. naive) ones. The proof of 1) allows us to conclude that preferred sets are liberal stable.

6) Consider the simple case where  $\mathcal{L} = \{a, b, x, p, \neg p\}$ ,  $\mathcal{R} = \{\frac{x}{p}, \frac{b}{\neg p}, \frac{a,b}{\neg p}\}$ ,  $\mathcal{A} = \{a, b, x\}$  and  $\mathcal{C} = \{\{x\}, \{\neg p\}, \{b\}, \{p\}, \{(a, b), \{p\}\}\}$ . In this case, A = {a} is complete but is not liberal stable.

7) In the same framework as in 6), note that all complete sets must include {a} (which is not attacked) and therefore A = {a} is minimally complete (grounded). However, A is not liberal stable.

4) Semi-stable sets must be complete. In the same framework as in 6), all complete sets contain a. So, {x} is neither complete nor semi-stable. However, {x} is liberal stable.

8) In the case where  $\mathcal{L} = \{a, b, c, p, \neg p\}$ ,  $\mathcal{R} = \{\frac{a,b}{p}, \frac{c}{\neg p}\}$ ,  $\mathcal{A} = \{a, b, c\}$  and  $\mathcal{C} = \{\{(a, b), \neg p\}, \{(c), p\}\}$ , the preferred sets collapse to the naive ones by symmetry, so the preferred sets are {a, b}, {c, a} and {c, b}. Ideal sets must be admissible and contained in the intersection  $\{a, b\} \cap \{c, a\} \cap \{c, b\} = \emptyset$ . So, the only ideal set is {} but none of the liberal stable sets is empty.

## 6. Summary and future work

Practical situations of decision making under strict uncertainty may be described qualitatively using basic influence diagrams. These diagrams give a logical structure to the decision domain and reveal the decision maker's most fundamental uncertainties. Basic influence diagrams can best be analysed and resolved using assumption-based argumentation. In the argumentation framework derived from a diagram and under a quite natural hypothesis called extensibility, we have proved that the possible outcomes of decisions are in one-to-one correspondence with the consequences of liberal stable sets of assumptions. Still under extensibility, we have shown that the set of liberal stable sets of assumptions includes all the naive, stable and preferred sets of assumptions and that it was included in the set of admissible sets of assumptions. In future work, we intend to develop an algorithm for the computation of liberal stable sets of assumptions and use this algorithm to transform basic influence diagrams into proper decision tables.

## Acknowledgements

This work was partially funded by the Sixth Framework IST programme of the European Commission, under the 035200 ARGUGRID project. The second author has also been supported by a UK Royal Academy of Engineering/Leverhulme Trust senior fellowship.

## References

- [Amgoud and Prade, 2004] L. Amgoud and H. Prade. Using arguments for making decisions: A possibilistic logic approach. In *Proceedings 20th Conference of Uncertainty in AI*, pages 10–17, 2004.
- [Bondarenko et al., 1997] A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1–2):63–101, 1997.
- [Caminada and Amgoud, 2007] M. Caminada and L. Amgoud. An axiomatic account of formal argumentation. *Artificial Intelligence*, 2007.
- [Caminada, 2006] M. Caminada. Semi-stable semantics. In *Proceedings of 1st International Conference on Computational Models of Argument*, 2006.
- [Coste-Marquis et al., 2005] C. Coste-Marquis, C. Devred, and P. Marquis. Symmetric argumentation frameworks. In *Proceedings 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 317–328, 2005.
- [Dung et al., 2002] P. M. Dung, P. Mancarella, and F. Toni. Argumentation-based proof procedures for credulous and sceptical non-monotonic reasoning. In *Computational Logic: Logic Programming and Beyond, Essays in Honour of R. A. Kowalski, Part II*, pages 289–310. Springer, 2002.
- [Dung et al., 2006] P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170(2):114–159, 2006.
- [Dung et al., 2007] P.M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence, Special Issue on Argumentation in Artificial Intelligence*, 171(10–15):642–674, 2007.
- [Dung, 1995] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence*, 77(2):321–257, 1995.
- [French, 1987] S. French. *Decision theory: an introduction to the mathematics of rationality*. Ellis Horwood, 1987.
- [Howard and Matheson, 1981] R. A. Howard and J. E. Matheson. Influence diagrams. In *Readings on the Principles and Applications of Decision Analysis*, volume II, pages 721–762. 1981.
- [Morge and Mancarella, 2007] M. Morge and P. Mancarella. The hedgehog and the fox. An argumentation-based decision support system. In *Proceedings 4th International Workshop on Argumentation in Multi-Agent Systems*, 2007.
- [Pearl, 1986] J. Pearl. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.
- [Takahashi and Sawamura, 2004] T. Takahashi and H. Sawamura. A logic of multiple-valued argumentation. In *Proceedings 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 800–807, 2004.
- [Toni, 2007] F. Toni. Assumption-based argumentation for closed and consistent defeasible reasoning. In *Proceedings 1st International Workshop on Juris-informatics in association with the 21st Annual Conference of The Japanese Society for Artificial Intelligence*, 2007.

## Appendix

**Lemma 2** Given a standard assumption-based argumentation framework and a set  $A$  of assumptions

1.  $A$  is stable in the standard sense (see [Bondarenko et al., 1997]) iff  $A$  is stable in the sense of section 2
2.  $A$  is complete in the standard sense (see [Bondarenko et al., 1997]) iff  $A$  is complete in the sense of section 2

**Proof 7** 1a) If  $A$  is stable in the standard sense, then  $A$  is conflict-free. If  $\neg(A \supseteq B)$  then  $\exists b \in B - A$ .  $b \notin A$ ,  $A$  attacks  $\{b\}$  and therefore  $A$  attacks  $B$ . Hence,  $A$  is stable in the new sense. 1b) If  $A$  is stable in the new sense, then  $A$  is conflict-free. If  $x \notin A$  then  $\neg(A \supseteq \{x\})$  so  $A$  attacks  $\{x\}$ .  $A$  is stable in the standard sense. 2a) If  $A$  is complete in the standard sense, then  $A$  is admissible. Assume  $B$  is defended by  $A$  and let  $b \in B$ . If  $\{b\}$  is not attacked, then  $A$  defends  $\{b\}$  and therefore  $b \in A$ . Otherwise,  $\{b\}$  is attacked.  $A$  defends  $B$  so  $A$  defends  $\{b\}$  and by completeness  $b \in A$ . So,  $A \supseteq B$ .  $A$  is complete in the new sense. 2b) If  $A$  is complete in the new sense, then  $A$  is admissible. If  $A$  defends  $\{x\}$ , then  $A \supseteq \{x\}$ , i.e.  $x \in A$ .  $A$  is complete in the standard sense.

# Integrating Object and Meta-Level Value Based Argumentation

Sanjay Modgil<sup>a</sup><sup>1</sup> Trevor Bench-Capon<sup>b</sup>

<sup>a</sup>Department of Computer Science, Kings College London

<sup>b</sup>Department of Computer Science, University of Liverpool

**Abstract.** A recent extension to Dung's argumentation framework allows for arguments to express preferences between other arguments. Value based argumentation can be formalised in this extended framework, enabling meta-level argumentation about the values that arguments promote, and the orderings on these values. In this paper, we show how extended frameworks integrating meta-level reasoning about values can be rewritten as Dung frameworks, and show a soundness and completeness result with respect to the rewrites. We then describe how value orderings can emerge, or be 'formed', as a result of dialogue games based on the rewritten frameworks, and illustrate the advantages of this approach over existing dialogue games for value based argumentation frameworks.

## 1. Introduction

A Dung argumentation framework [5] consists of a set of arguments related by a binary conflict based attack relation. A 'calculus of opposition' is then applied to determine the sets of acceptable arguments under different extensional semantics. The framework abstracts from the underlying logic in which the arguments and attack relation are defined. Dung's theory has thus become established as a general framework for various species of non-monotonic reasoning, and, more generally, reasoning in the presence of conflict.

The extensional semantics may yield multiple sets of acceptable arguments (extensions). The sceptically justified arguments are those that appear in every extension. However, one may then be faced with the problem of how to choose between arguments that belong to at least one, but not all extensions (the credulously justified arguments), when they conflict. One solution is to provide some means for preferring one argument to another so that one can then determine whether attacks succeed and defeat the arguments they attack. For example, Value Based Argumentation Frameworks (VAFs) [2] associate each argument with a social value which it promotes, and this property determines the strength of arguments by reference to an ordering on these social values. Given such a preference ordering, one obtains a defeat relation with respect to that ordering. It has been shown that if a framework does not contain cycles comprising only arguments of equal strength, then on the basis of the defined defeat relation obtained with respect to a given ordering, one can obtain a unique, non-empty extension under the *preferred* semantics.

---

<sup>1</sup>This author is supported by the EU 6th Framework project CONTRACT (INFSO-IST-034418). The opinions expressed herein are those of the named authors only and should not be taken as necessarily representative of the opinion of the European Commission or CONTRACT project partners.

However, in general, preference information is often itself defeasible, conflicting and so may itself be subject to argumentation based reasoning. Hence, Dung's framework has recently been extended [7,8] to include arguments that claim preferences between other arguments. Specifically, the framework is extended to include a second attack relation such that an argument expressing a preference between two other arguments, attacks the binary attack between these two conflicting arguments, thus determining which attacks succeed as defeats. Arguments expressing contradictory preferences attack each other, and one can then argue over which of these ‘preference arguments’ is preferred to, and so defeats, the other. The justified arguments of an *Extended Argumentation Framework* (*EAF*) can then be evaluated under the full range of Dung's extensional semantics. Examples of value based argumentation in the extended semantics have informally been described in [7,8], whereby different value orderings may yield contradictory preferences, requiring meta-level reasoning *about* values and value orderings to determine a unique set of justified arguments.

In this paper the extended semantics are reviewed in section 2. Sections 3 and 4 then describe the main contributions of this paper:

- 1) We formalise *EAF*'s integrating meta-level reasoning about values and value orderings, and then show that such *EAF*s can be rewritten as Dung argumentation frameworks. We show a soundness and completeness result for the rewrite.
- 2) Given 1), we can then exploit results and techniques applied to Dung argumentation frameworks. In particular, we show how value orderings can emerge from dialogue games based on the above rewrites, and demonstrate the advantages of this approach over games proposed specifically for *VAFs* [3].

## 2. Extended Argumentation Frameworks

A Dung argumentation framework (*AF*) [5] is of the form  $(\text{Args}, \mathcal{R})$  where  $\mathcal{R} \subseteq (\text{Args} \times \text{Args})$  can denote either attack or defeat. An argument  $A \in \text{Args}$  is defined as acceptable w.r.t. some  $S \subseteq \text{Args}$ , if for every  $B$  such that  $(B, A) \in \mathcal{R}$ , there exists a  $C \in S$  such that  $(C, B) \in \mathcal{R}$ . Intuitively,  $C$  ‘reinstates’  $A$ . In [5], the acceptability of a set of arguments under different extensional semantics is then defined. The definition of admissible and preferred semantics are given here, in which  $S \subseteq \text{Args}$  is conflict free if no two arguments in  $S$  are related by  $\mathcal{R}$ .

**Definition 1** Let  $S \subseteq \text{Args}$  be a conflict free set. Then  $S$  is admissible iff each argument in  $S$  is acceptable w.r.t.  $S$ .  $S$  is a preferred extension iff  $S$  is a set inclusion maximal admissible extension.

From hereon, an argument is said to be credulously, respectively sceptically, justified, iff it belongs to at least one, respectively all, preferred extensions. We now present the extended argumentation semantics described in [7,8]. By way of motivation, consider two individuals **P** and **O** exchanging arguments  $A, B \dots$  about the weather forecast:

- P** : “Today will be dry in London since the BBC forecast sunshine” =  $A$
- O** : “Today will be wet in London since CNN forecast rain” =  $B$
- P** : “But the BBC are more trustworthy than CNN” =  $C$
- O** : “However, statistics show that CNN are more accurate than the BBC” =  $C'$
- O** : “And basing a comparison on statistics is more rigorous and rational than basing a comparison on your instincts about their relative trustworthiness” =  $E$

Arguments  $A$  and  $B$  symmetrically attack, i.e.,  $(A, B), (B, A) \in \mathcal{R}$ .  $\{A\}$  and  $\{B\}$  are admissible. We then have argument  $C$  claiming that  $A$  is preferred to  $B$ . Hence  $B$  does not successfully attack (defeat)  $A$ , but  $A$  does defeat  $B$ . Intuitively,  $C$  is an argument for

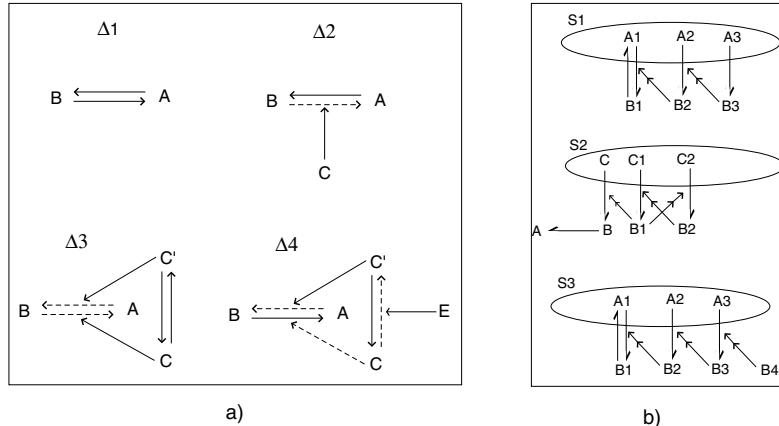


Figure 1.

$A$ 's repulsion of, or defence against,  $B$ 's attack on  $A$ , i.e.,  $C$  **attacks**  $B$ 's attack on  $A$  ( $\Delta_2$  in figure 1a)) so that  $B$ 's attack on  $A$  does not succeed as a defeat.  $B$ 's attack on  $A$  is ‘cancelled out’, and we are left with  $A$  defeating  $B$ . Only  $\{A\}$  is now admissible. Of course, given  $C'$  claiming a preference for  $B$  over  $A$  and so attacking  $A$ 's attack on  $B$ , then we will have that  $\{A\}$  and  $\{B\}$  are now both admissible, since neither defeats the other.  $C$  and  $C'$  claim contradictory preferences and so attack each other ( $\Delta_3$  in figure 1a)). These attacks can themselves be subject to attacks in order to determine the defeat relation between  $C$  and  $C'$  and so  $A$  and  $B$ .  $E$  attacks the attack from  $C$  to  $C'$  ( $\Delta_4$  in figure 1a)), and so determines that  $C'$  defeats  $C$ ,  $B$  defeats  $A$ , and the discussion concludes in favour of  $\mathbf{O}$ 's argument that it will be a wet day in London. We now formally define the elements of an *Extended Argumentation Framework*, and the defeat relation that is now parameterised w.r.t. some set  $S$  of arguments.

**Definition 2** An *Extended Argumentation Framework* (EAF) is a tuple  $(\text{Args}, \mathcal{R}, \mathcal{D})$  such that  $\text{Args}$  is a set of arguments, and:

- $\mathcal{R} \subseteq \text{Args} \times \text{Args}$
- $\mathcal{D} \subseteq (\text{Args} \times \mathcal{R})$
- If  $(C, (A, B)), (C', (B, A)) \in \mathcal{D}$  then  $(C, C'), (C', C) \in \mathcal{R}$

**Notation 1** We may write  $A \rightarrow B$  to denote  $(A, B) \in \mathcal{R}$ . If in addition  $(B, A) \in \mathcal{R}$ , we may write  $A \rightleftharpoons B$ . We may also write  $C \rightarrow (A \rightarrow B)$  to denote  $(C, (A, B)) \in \mathcal{D}$

**Definition 3**  $A$  defeats $_S$   $B$ , denoted by  $A \rightarrow^S B$ , iff  $(A, B) \in \mathcal{R}$  and  $\neg \exists C \in S$  s.t.  $(C, (A, B)) \in S$ .

Referring to the weather forecast example,  $A$  defeats $_\emptyset$   $B$  but  $A$  does not defeat $_{\{C'\}}$   $B$  ( $A \rightarrow^{\{C'\}} B$ ). The notion of a conflict free set  $S$  of arguments is now defined so as to

account for the case where an argument  $A$  asymmetrically attacks  $B$ , but given a preference for  $B$  over  $A$ , both may appear in a conflict free set and hence an extension (as in the case of value based argumentation).

**Definition 4**  $S$  is conflict free iff  $\forall A, B \in S$ : if  $(B, A) \in \mathcal{R}$  then  $(A, B) \notin \mathcal{R}$ , and  $\exists C \in S$  s.t.  $(C, (B, A)) \in \mathcal{D}$ .

We now define the acceptability of an argument  $A$  w.r.t. a set  $S$  for an *EAF*. The definition is motivated in more detail in [7,8] and relates to an intuitive requirement (captured by Dung's fundamental lemma in [5]) on what it means for an argument to be acceptable w.r.t. an admissible set  $S$  of arguments: *if  $A$  is acceptable with respect to  $S$ , then  $S \cup \{A\}$  is admissible*. To ensure satisfaction of this requirement, acceptability for *EAFs* requires the notion of a *reinstatement set* for a defeat.

**Definition 5** Let  $S \subseteq \text{Args}$  in  $(\text{Args}, \mathcal{R}, \mathcal{D})$ . Let  $R_S = \{X_1 \rightarrow^S Y_1, \dots, X_n \rightarrow^S Y_n\}$  where for  $i = 1 \dots n$ ,  $X_i \in S$ . Then  $R_S$  is a reinstatement set for  $C \rightarrow^S B$ , iff:

- $C \rightarrow^S B \in R_S$ , and
- $\forall X \rightarrow^S Y \in R_S, \forall Y' \text{ s.t. } (Y', (X, Y)) \in \mathcal{D}, \exists X' \rightarrow^S Y' \in R_S$

**Definition 6**  $A$  is acceptable w.r.t.  $S \subseteq \text{Args}$  iff  $\forall B \text{ s.t. } B \rightarrow^S A, \exists C \in S \text{ s.t. } C \rightarrow^S B$  and there is a *reinstatement set* for  $C \rightarrow^S B$ .

In figure 1b),  $A1$  is acceptable w.r.t.  $S1$ . We have  $B1 \rightarrow^{S1} A1$  and  $A1 \rightarrow^{S1} B1$ . The latter is based on an attack that is attacked by  $B2$ . However,  $A2 \rightarrow^{S1} B2$ , which in turn is challenged by  $B3$ . But then,  $A3 \rightarrow^{S1} B3$ . We have the reinstatement set  $\{A1 \rightarrow^{S1} B1, A2 \rightarrow^{S1} B2, A3 \rightarrow^{S1} B3\}$  for  $A1 \rightarrow^{S1} B1$ . Note that  $A$  is acceptable w.r.t.  $S2$  given the reinstatement set  $\{C \rightarrow^{S2} B, C1 \rightarrow^{S2} B1, C2 \rightarrow^{S2} B2\}$  for  $C \rightarrow^{S2} B$ . Finally  $A1$  is not acceptable w.r.t  $S3$  since no argument in  $S3$  defeats <sub>$S3$</sub>   $B4$ .

Admissible and preferred semantics for *EAFs* are now given by definition 1, where conflict free is defined as in definition 4. (Dung's definition of complete, stable and grounded semantics also apply to *EAFs* [7,8]). In our weather example,  $\{B, C', E\}$  is the single preferred extension. In [7,8] we show that *EAFs* inherit many of the fundamental results holding for extensions of a Dung framework. In particular: a) If  $S$  is admissible and arguments  $A$  and  $A'$  are acceptable w.r.t.  $S$ , then  $S \cup \{A\}$  is admissible and  $A'$  is acceptable w.r.t.  $S \cup \{A\}$ ; b) the set of all admissible extensions of an *EAF* forms a complete partial order w.r.t. set inclusion; c) for each admissible  $S$  there exists a preferred extension  $S'$  such that  $S \subseteq S'$ ; d) Every *EAF* possesses at least one preferred extension.

### 3. Value Based Argumentation in Extended Argumentation Frameworks

In this section we show how meta-level argumentation about values and value orderings can be captured in a special class of *EAFs*. We then show that these *EAFs* can be rewritten as Dung argumentation frameworks, and go on to show a soundness and completeness result with the original *EAFs*.

**Definition 7** A value-based argumentation framework (*VAF*) is a 5-tuple  $\langle \text{Args}, \mathcal{R}, V, \text{val}, P \rangle$  where  $\text{val}$  is a function from  $\text{Args}$  to a non-empty set of values  $V$ , and  $P$  is a set  $\{a_1, \dots, a_n\}$ , where each  $a_i$  names a total ordering (audience)  $>_{a_i}$  on  $V \times V$ .

An audience specific VAF (*aVAF*) is a 5-tuple  $\langle \text{Args}, \mathcal{R}, V, \text{val}, a \rangle$  where  $a \in P$ .

Given an *aVAF*  $\Gamma = \langle \text{Args}, \mathcal{R}, V, \text{val}, a \rangle$ , one can then say that  $A \in \text{Args}$  defeats<sub>*a*</sub>  $B \in \text{Args}$ , if  $(A, B) \in \mathcal{R}$  and it is not the case that  $\text{val}(B) >_a \text{val}(A)$ . Letting  $\mathcal{R}_a$  denote the binary relation defeats<sub>*a*</sub>, then the extensions and justified arguments of  $\Gamma$  are now those of the framework  $(\text{Args}, \mathcal{R}_a)$  (as defined in definition 1). If for every  $(A, B) \in \mathcal{R}$  either  $\text{val}(A) >_a \text{val}(B)$  or  $\text{val}(B) >_a \text{val}(A)$ , and assuming no cycles in the same value in  $\Gamma$  (no cycle in the argument graph whose contained arguments promote the same value) then there is guaranteed to be a unique, non-empty preferred extension of  $\Gamma$ , and a polynomial time algorithm to find it [2].

Pairwise orderings on values can be interpreted as *value preference arguments* in an *EAF*. Consider the mutually attacking arguments  $A$  and  $B$ , respectively promoting values  $v_1$  and  $v_2$ . Then  $v_1 > v_2$  can be interpreted as a *value preference argument* expressing that  $A$  is preferred to  $B$  -  $(v_1 > v_2) \rightarrow (B \dashv A)$  - and  $v_2 > v_1$  as expressing the contrary preference (see figure 2a)). The choice of audience can then be expressed as an *audience argument* that attacks the attacks between the value preference arguments. Suppose the argument  $v_1|v_2$  denoting the chosen audience that orders  $v_1 > v_2$ . Then  $v_1|v_2 \rightarrow ((v_2 > v_1) \rightharpoonup (v_1 > v_2))$ . Now the unique preferred extension of the *EAF* in figure 2a) is  $\{A, v_1 > v_2, v_1|v_2\}$ . In this way, we can represent the meta-level reasoning required to find the preferred extension of an *aVAF*.

**Definition 8** Let  $\Gamma$  be an *aVAF*  $\langle \text{Args}, \mathcal{R}, V, \text{val}, a \rangle$ . Then the *EAF*  $\Delta = (\text{Args}1 \cup \text{Args}2 \cup \text{Args}3, \mathcal{R}1 \cup \mathcal{R}2 \cup \mathcal{R}3, \mathcal{D}1 \cup \mathcal{D}2 \cup \mathcal{D}3)$  is defined as follows:

1.  $\text{Args}1 = \text{Args}, \mathcal{R}1 = \mathcal{R}$
2.  $\{v > v' | v, v' \in V, v \neq v'\} \subseteq \text{Args}2^2$ ,  
 $\{(v > v', v' > v) | v > v', v' > v \in \text{Args}2\} \subseteq \mathcal{R}2$
3.  $\{a\} \subseteq \text{Args}3, \emptyset \subseteq \mathcal{R}3$
4.  $\{ (v > v', (A, B)) | (A, B) \in \mathcal{R}1, \text{val}(B) = v, \text{val}(A) = v' \} \subseteq \mathcal{D}1$   
 $\{ (a, (v > v', v' > v)) | a \in \text{Args}3, (v > v', v' > v) \in \mathcal{R}2, v' >_a v \} \subseteq \mathcal{D}2$   
 $\mathcal{D}3 = \emptyset$

If in 2,3 and 4, the  $\subseteq$  relation is replaced by  $=$ , then  $\Gamma$  and  $\Delta$  are said to be *equivalent*.

If an *aVAF*  $\Gamma$  and the defined *EAF*  $\Delta$  are *equivalent*, one can straightforwardly show that for any  $A \in \text{Args}$  in  $\Gamma$ ,  $A$  is a sceptically, respectively credulously, justified argument of  $\Gamma$  iff  $A$  is a sceptically, respectively credulously, justified argument of  $\Delta$ .

Notice that  $\Delta$  is defined so that one could additionally consider other arguments and attacks in levels 2 and 3. For example, arguments in level 2 that directly attack *value preference arguments*, or arguments in level 3 representing different audiences. Notice also the hierarchical nature of the defined *EAF*  $\Delta$ . It is stratified into three levels such that binary attacks are between arguments within a given level, and defence attacks originate from arguments in the immediate meta-level. In general then, incorporating meta-level argumentation about values and value orderings can be modelled in hierarchical *EAFs*<sup>3</sup>.

<sup>2</sup>Note that this adds an additional  $|V|(|V| - 1|)$  arguments: this, however, is acceptable, since it is only polynomial in the number of values.

<sup>3</sup>See [7,8] for examples illustrating requirements for *EAFs* that do not ‘stratify’ the argumentation in this way.

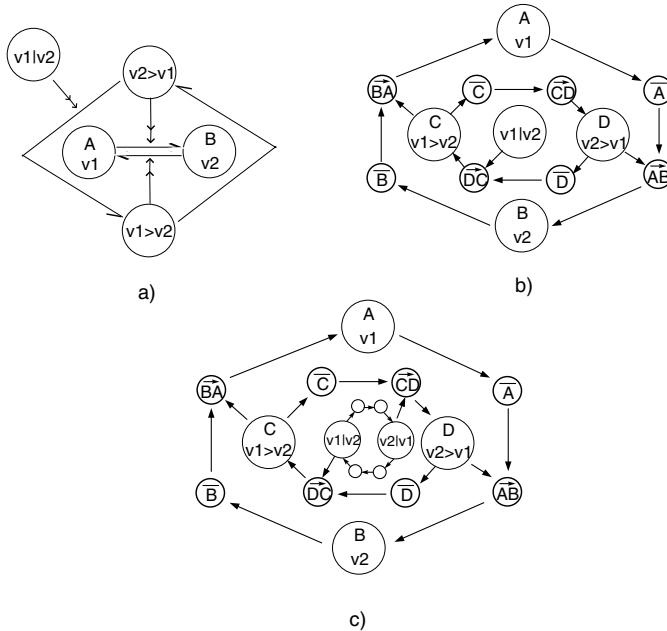


Figure 2.

**Definition 9**  $\Delta = (\text{Args}, \mathcal{R}, \mathcal{D})$  is a hierarchical EAF iff there exists a partition  $\Delta_H = ((\text{Args}_1, \mathcal{R}_1, \mathcal{D}_1), \dots, (\text{Args}_n, \mathcal{R}_n, \mathcal{D}_n))$  such that:

- $\text{Args} = \bigcup_{i=1}^n \text{Args}_i$ ,  $\mathcal{R} = \bigcup_{i=1}^n \mathcal{R}_i$ ,  $\mathcal{D} = \bigcup_{i=1}^n \mathcal{D}_i$ , and for  $i = 1 \dots n$ ,  $(\text{Args}_i, \mathcal{R}_i)$  is a Dung argumentation framework.
- $\mathcal{D}_n = \emptyset$ , and for  $i = 1 \dots n - 1$ ,  $(C, (A, B)) \in \mathcal{D}_i$  implies  $(A, B) \in \mathcal{R}_i$ ,  $C \in \text{Args}_{i+1}$

We now show that it is possible to rewrite a hierarchical EAF  $\Delta$  as a Dung argumentation framework  $AF_\Delta$ , such that the preferred extensions of  $\Delta$  and  $AF_\Delta$  are equivalent modulo the additional arguments included in the rewrite. Firstly, we define an *expansion* of a Dung framework in which each attack  $(A, B)$  is replaced by a set of attacks  $\{(A, \overline{A}), (\overline{A}, \overline{AB}), (\overline{AB}, B)\}$ , and the additional arguments  $\overline{A}$  and  $\overline{AB}$  are included. Intuitively,  $\overline{A}$  stands for “ $A$  is not acceptable”, and  $\overline{AB}$  stands for “ $A$  defeats  $B$ ”.

**Definition 10** Let  $AF = (\text{Args}, \mathcal{R})$  be a Dung argumentation framework. Then the *expansion* of  $AF$  is the framework  $AF' = (\text{Args}', \mathcal{R}')$ , where:

- $\mathcal{R}' = \bigcup_{(X,Y) \in \mathcal{R}} \{(X, \overline{X}), (\overline{X}, \overline{XY}), (\overline{XY}, Y)\}$
- $\text{Args}' = \text{Args} \cup \bigcup_{(X,Y) \in \mathcal{R}} \{\overline{X}, \overline{XY}\}$

Proposition 1 below follows immediately from lemma 1 in the Appendix.

**Proposition 1** Let  $AF' = (Args', \mathcal{R}')$  be the expansion of  $AF = (Args, \mathcal{R})$ . Then  $A \in Args$  is a sceptically, respectively credulously, justified argument of  $AF$  iff  $A$  is a sceptically, respectively credulously, justified argument of  $AF'$ .

We now formally define the rewrite of an *EAF* as a Dung argumentation framework:

**Definition 11** Let  $\Delta = (Args, \mathcal{R}, \mathcal{D})$ . Let  $(Args', \mathcal{R}')$  be the expansion of  $(Args, \mathcal{R})$ . Then  $AF_\Delta = (Args_\Delta, \mathcal{R}_\Delta)$  where:

- $Args_\Delta = Args'$
- $\mathcal{R}_\Delta = \mathcal{R}' \cup \{(C, \overrightarrow{AB}) \mid (C, (A, B)) \in \mathcal{D}\}$ .

Figure 2b) shows the rewrite of the *EAF* in figure 2a). The single preferred extension of the *EAF* is  $\{A, v_1 > v_2, v_1|v_2\}$ , and (where  $C = v_1 > v_2, D = v_2 > v_1$ ) the single preferred extension of the rewrite is  $\{A, \overrightarrow{AB}, \overrightarrow{B}, C, \overrightarrow{CD}, \overrightarrow{D}, v_1|v_2\}$ . Theorem 1 follows immediately from lemma 2 in the Appendix.

**Theorem 1** Let  $\Delta = (Args, \mathcal{R}, \mathcal{D})$  be a hierarchical *EAF*.  $A \in Args$  is a sceptically, respectively credulously, justified argument of  $\Delta$ , iff  $A$  is a sceptically, respectively credulously, justified argument of the rewrite  $AF_\Delta$ .

As mentioned earlier, one might additionally include more than one audience argument in level 3 of an *EAF*. Given a *VAF*  $\langle Args, \mathcal{R}, V, val, P \rangle$ , then its *EAF* is obtained as in definition 8, except that now  $\{a \mid a \in P\} \subseteq Args_3$  (recall that  $P$  is the set of all possible audiences). If  $\{a \mid a \in P\} = Args_3$ , then we say that the *VAF* and its obtained *EAF* are *equivalent*. Notice that if for any  $a, a' \in P$ ,  $(a, (v > v', v' > v)), (a', (v' > v, v > v')) \in \mathcal{D}_2$ , then it follows from the definition of an *EAF* (definition 2) that  $a$  and  $a'$  attack each other, i.e.  $(a, a'), (a', a) \in \mathcal{R}_3$ . Figure 2c) shows the rewrite of such an *EAF* as an *AF*, extending the example in figure 2b) to include the alternative audience choice.

Recall that each possible audience argument corresponds to a different total orderings on the values. Also,  $\forall v, v' \in V, v > v'$  and  $v' > v$  are value preference arguments in *Args*. Hence, every audience argument will attack every other audience argument. Moreover, each audience argument will give rise to a corresponding preferred extension (under the assumption of no cycles in the same value), so that we no longer have a unique preferred extension.

None the less, there are some nice properties of the *AF* in figure 2c). In [2], the arguments that appear in the preferred extension for every, respectively at least one, audience, are referred to as *objectively*, respectively *subjectively*, acceptable. One can straight-forwardly show that:

*A is an objectively, respectively subjectively, acceptable argument of a *VAF* iff A is a sceptically, respectively credulously, justified argument of the *AF* rewrite of the *VAF*'s equivalent *EAF*.*

Moreover our task is bounded in that all the preferred extensions depend on a single choice of audience argument. However, if all possible audience arguments are included, then there are  $|V|$  factorial many audience arguments, rendering the augmented Dung graph impractical for even moderately large number of values. None the less, for small values of  $|V|$  there may be utility in presenting the complete picture in the manner shown in figure 2c). Moreover, choice of an audience can be effected through submission of other arguments attacking audience arguments, or indeed ascending to level 4 to argue about preferences between audiences (as described in [7]).

#### 4. Emergence of Value Orderings as a Product of Reasoning

We have thus far assumed that value orderings are available at the outset. However, as Searle [9] points out, this is unrealistic; preference orderings are more often the product of practical reasoning rather than an input to it. This has motivated development of dialogue games [3,1] in which value orderings emerge from the attempts of a proponent to defend an argument, in much the same manner as [4] used a dialogue game to establish the preferred extension of an *AF*. We illustrate use of these games using a three cycle in two values as shown in Figure 3a).

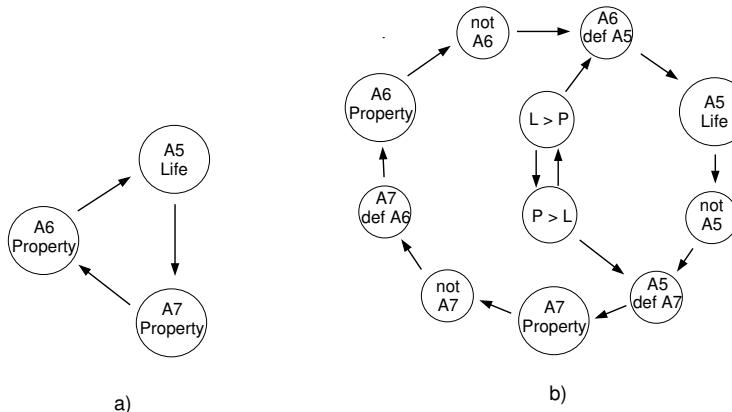


Figure 3.

Consider this first as an *AF*. [4]'s *TPI* (Two Party Immediate Dispute) game begins with Proponent (Prop) choosing an argument he wishes to defend by showing that it is included in some preferred extension. Suppose Prop chooses *A5*. Since in an *AF* attacks always succeed, this means that any argument attacked by a chosen argument - in this case *A7* - cannot be in the same preferred extension, and so cannot be used by Prop subsequently. The game proceeds by the Opponent (Opp) playing an argument - *A6* - which attacks the argument played by Prop'. Now the game ends with a victory for Opp, because the attacker *A7* of *A6* is not available to Prop. Thus *A5* cannot be incorporated in a preferred extension. This is as it should be: in an *AF* the preferred extension of a three cycle is empty. In a *VAF*, however, there are two preferred extensions of the three cycle shown in figure 3a), depending on whether Life (L) is preferred to Property (P) or vice versa. If  $L > P$  the preferred extension is  $\{A5, A6\}$ , and if  $P > L$  it is  $\{A5, A7\}$ . Note that *A5* is objectively accepted. In [1]'s *VTPI* (Value based *TPI*) game, *TPI* is augmented to enable Prop to defend an argument by claiming a preference for the value of the attacked argument over its attacker. Thus if Prop chooses *A5*, and Opp plays *A6*, Prop may defend *A5* by claiming  $L > P$ . Under *VTPI*, *A7* is also unavailable to Prop because it is attacked by the originally chosen argument *A5*. There is a major problem with this: since *A5* is objectively acceptable, it should not have been necessary for Prop to commit to a value preference in order to defend it. The objective acceptability of *A5* is lost. This might be remedied by keeping *A7* available. However this requires that Prop,

**Table 1.** Game establishing objective acceptance of  $A_5$ 

Play	Prop Moves	Opp Moves	Prop Commitments	Opp Commitments
1	$A_5$	$A_6 \text{ def } A_5$	$A_5$	$A_6 \text{ def } A_5$
2	$L > P$	$P > L$	$A_5, L > P$	$A_6 \text{ def } A_5, P > L$
3	RETRACT $L > P:$ $\text{not } A_6$	$A_6$	$A_5, \text{not } A_6$	$A_6 \text{ def } A_5, P > L, A_6$
4	$A_7 \text{ def } A_6$	$\text{not } A_7$	$A_5, \text{not } A_6, A_7 \text{ def } A_6$	$A_6 \text{ def } A_5, P > L, A_6, \text{not } A_7$
5	$A_7$		$A_5, \text{not } A_6, A_7 \text{ def } A_6, A_7$	$A_6 \text{ def } A_5, P > L, A_6, \text{not } A_7$

when playing  $A_5$ , declares that  $A_7$  is not defeated by  $A_5$ , which in turn forces Prop to commit to  $P > L$ . Now Prop can use  $A_7$  to defend  $A_5$  against  $A_6$ , but the damaging commitment has already been made. Moreover this greatly complicates the game, as is shown by the game presented in [3], where a similar move requires a distinction between attacks, successful attacks and definite attacks, in order to keep arguments available for subsequent use. The game in [3] also suffers the problem of being forced to choose a particular value order to defend even an objectively acceptable argument.

We now informally describe a game w.r.t. a *partial* rewrite of a VAF's EAF. The rewrite is partial in the sense that only value preference arguments are included, and attacks between these arguments are not expanded as described in definition 10. Consider now the three cycle partial rewrite in figure 3b) (in which we write  $\text{not } X$  and  $X \text{ def } Y$  instead of  $\overline{X}$  and  $\overline{X}Y$ ). Since this is effectively an AF, we can use the *TPI* game to find a preferred extension. In *TPI*, a player who cannot defend an argument against an attack can retract the argument if there is another line of defense which does not use it. Now consider Prop's defense of  $A_5$ . Choosing  $A_5$  now has no effect on  $A_7$ : it only excludes the argument  $\text{not } A_5$ , which seems intuitively correct. Importantly therefore  $A_7$  will remain available for Prop's later use. Prop has a choice of two moves to defend  $A_5$ :  $L > P$  and  $\text{not } A_6$ . Suppose Prop plays  $L > P$ : although this appears to be a commitment to a value ordering this will only be temporary. Opp has no choice but to attack this with  $P > L$ . Prop cannot defend against this and so must retract and so abandon his commitment to  $L > P$ , and pursue the alternative defence of  $A_5$ . Since this retraction is conditional on  $P > L$ , Opp remains committed to  $P > L$ . Now the game continues as shown in Table 1. At Play 5 Prop will win, since  $A_5 \text{ def } A_7$  is no longer available to Opp, as Opp is committed to an attacker  $P > L$ . Nor is Opp able to retract the commitment to  $P > L$ , since there is no other way to attack  $L > P$ , i.e., there is no alternative defence available to Opp, so backtracking is not possible. Thus Prop is able to defend  $A_5$  without embracing any commitment to value preferences: the only commitment to a preference when the game is complete, is on the part of Opp.

Similar advantages accrue if we consider the defence of a subjectively acceptable argument such as  $A_6$ . This is defensible only for audiences with  $L > P$ , so that  $A_6$  does not defeat  $A_5$ , allowing  $A_5$  to defeat  $A_7$ . But in *VTPI*,  $A_5$  is unavailable once we have included  $A_6$ . However use of *TPI* for the framework in figure 3b) now gives rise to the game in Table 2. Here, playing  $A_6$  does not exclude  $A_5$ , but only  $\text{not } A_6$ . Prop is forced to commit to  $L > P$ , but this also enables Prop to defend against  $A_6 \text{ def } A_5$ , whereupon Opp cannot play the already retracted  $P > L$ . Thus  $A_6$  is defensible for the audience which prefers  $L$  to  $P$ , although note that Opp is not obliged to agree with this value preference.

Another advantage of the augmented framework is that the explicit representation of value preferences ensures that commitments to value preferences are propagated im-

**Table 2.** Game establishing subjective acceptance of  $A_6$ 

Play	Prop Moves	Opp Moves	Prop Commitments	Opp Commitments
1	$A_6$	$A_7 \text{ def } A_6$	$A_6$	$A_7 \text{ def } A_6$
2	$\text{not } A_7$	$A_7$	$A_6, \text{not } A_7$	$A_7 \text{ def } A_6, A_7$
3	$A_5 \text{ def } A_7$	$P > L$	$A_6, \text{not } A_7, A_5 \text{ def } A_7$	$A_7 \text{ def } A_6, A_7, P > L$
4	$L > P$	RETRACT $P > L:$ $\text{not } A_5$	$A_6, \text{not } A_7, A_5 \text{ def } A_7$ $L > P$	$A_7 \text{ def } A_6, A_7, \text{not } A_5$
5	$A_5$	$A_6 \text{ def } A_5$	$A_6, \text{not } A_7, A_5 \text{ def } A_7,$ $L > P, A_5$	$A_7 \text{ def } A_6, A_7, \text{not } A_5,$ $A_6 \text{ def } A_5$
6	$L > P$		$A_6, \text{not } A_7, A_5 \text{ def } A_7,$ $L > P, A_5$	$A_7 \text{ def } A_6, A_7, \text{not } A_5,$ $A_6 \text{ def } A_5$

mediately, instead of requiring potentially complicated housekeeping to ensure that the emerging value order is consistent. Because an argument of the form  $L > P$  attacks (and defeats) every argument of the form  $A \text{ def } B$ , making this commitment in respect of one argument anywhere in the framework will protect all other arguments which are rendered admissible by this value preference.

Finally, note that while [3] suggests a defence at the object level should always be attempted before resorting to value preferences, as this has fewer ramifications elsewhere in the framework, this is less clearly the correct strategy to use with *TPI* played over our augmented framework. Thus, in the example in Table 1, it was tactically useful for Prop to first make a defence with a value preference. As this preference is only attacked by the opposite value preference, this forces the opponent to play this preference. Now when Prop retracts the value preference and makes the alternative defence, he is not committed to a particular audience, and can exploit the opponent's commitment to force acceptance of his argument.

## 5. Conclusions and Future Work

We have reviewed an extension of Dung's argumentation framework that enables integration of meta-level reasoning about which arguments should be preferred, and shown how certain useful cases, of which value based argumentation frameworks are an example, can be rewritten as a standard Dung framework. This enables results and techniques that apply to, and have been developed for, standard frameworks to be used directly for frameworks integrating meta-level reasoning about preferences in general, and values in particular. As an illustration of the advantages that accrue, we showed how a dialogue game devised for standard *AFs* can be used to identify the value order under which a particular argument can be defended. Now that arguments committing to a preference are of the same sort as other arguments in the framework, the problems arising from the need to give special treatment to these commitments in existing games can be avoided.

Future work will explore how our extended framework can assist in handling arguments which promote values to different degrees, previously treated in [6] and arguments which promote multiple values. In both cases this will provide an extra source of attacks on arguments of the form 'A defeats B'. For example, in current VAFs an attack by an argument promoting the same value always succeeds. However, allowing for different degrees of promotion will offer the possibility of the attacked argument defending itself by promoting the value to a greater degree than its attacker.

## 6. Appendix

**Lemma 1** Let  $AF' = (\text{Args}', \mathcal{R}')$  be the expansion of  $AF = (\text{Args}, \mathcal{R})$ . Then  $S \subseteq \text{Args}$  is an admissible extension of  $AF$  iff  $T \subseteq \text{Args}'$  is a admissible extension of  $AF'$ , where

$$T = S \cup \{\overrightarrow{XY} \mid Y \in S, (X, Y) \in \mathcal{R}\} \cup \{\overrightarrow{YZ} \mid Y \in S, (Y, Z) \in \mathcal{R}\}.$$

**Proof:** It is straightforward to show that  $S$  is conflict free iff  $T$  is conflict free. It remains to show that every argument in  $S$  is acceptable w.r.t.  $S$  iff every argument in  $T$  is acceptable w.r.t.  $T$ .

*Left to Right half:* Suppose some  $A \in S, (B, A) \in \mathcal{R}$ . By definition of  $AF'$ :

$$\forall A \in \text{Args}, (B, A) \in \mathcal{R} \text{ iff } (X, A) \in \mathcal{R}', \text{ where } X = \overrightarrow{BA} \text{ and } (\overrightarrow{B}, \overrightarrow{BA}) \in \mathcal{R}' \quad (1)$$

By definition of  $T$ ,  $\overrightarrow{B} \in T$ . Hence:

$$\forall A \in \text{Args}, A \in S \text{ is acceptable w.r.t. } S \text{ implies } A \in T \text{ is acceptable w.r.t. } T \quad (2)$$

We show that  $\overrightarrow{B}$  is acceptable w.r.t.  $T$ . By assumption of  $A$  is acceptable w.r.t.  $S$ ,  $\exists C \in S, (C, B) \in \mathcal{R}$ . By definition of  $T$ :  $C, \overrightarrow{CB} \in T$ . By definition of  $AF'$ ,  $(X, \overrightarrow{B}) \in \mathcal{R}'$  implies  $X = B$ , and  $(\overrightarrow{CB}, B) \in \mathcal{R}'$ . Hence  $\overrightarrow{B}$  is acceptable w.r.t.  $T$ .

We show that  $\forall A \in \text{Args}$  s.t.  $A \in T$ , if  $(A, C) \in \mathcal{R}$ , then  $\overrightarrow{AC}$  is acceptable w.r.t.  $T$ . This follows from the definition of  $AF'$ , where  $(X, \overrightarrow{AC}) \in \mathcal{R}'$  implies  $X = \overrightarrow{A}$ , and  $(A, \overrightarrow{A}) \in \mathcal{R}'$ .

*Right to Left half:* For any  $A \in \text{Args}$ ,  $A \in T$  we need to show that  $A$  is acceptable w.r.t.  $S$ . Suppose  $(X, A) \in \mathcal{R}'$ . By (1),  $X$  is some  $\overrightarrow{BA}$  s.t.  $(B, A) \in \mathcal{R}$ ,  $(\overrightarrow{B}, \overrightarrow{BA}) \in \mathcal{R}'$ , and by definition of  $T$ ,  $\overrightarrow{B} \in T$ .

By definition of  $AF'$ , if  $(X, \overrightarrow{B}) \in \mathcal{R}'$  then  $X = B$ . Since  $\overrightarrow{B} \in T$  and  $T$  is admissible,  $\exists \overrightarrow{CB} \in T$  s.t.  $(\overrightarrow{CB}, B) \in \mathcal{R}'$ , and so  $(C, B) \in \mathcal{R}$ .

If  $(X, \overrightarrow{CB}) \in \mathcal{R}'$ , then  $X = \overrightarrow{C}$ . If  $(Y, \overrightarrow{C}) \in \mathcal{R}'$  then  $Y = C$ , where  $C \in \text{Args}$ . Hence, since  $\overrightarrow{CB}$  is acceptable w.r.t.  $T$ , it must be that  $C \in T$ . By assumption,  $C \in S$ . Hence  $A$  acceptable w.r.t.  $S$ .

Proof of lemma 2 makes use of the following partition of a hierarchical EAF's rewrite:

**Definition 12** Let  $\Delta_H = ((\text{Args}_1, \mathcal{R}_1), \mathcal{D}_1), \dots, ((\text{Args}_n, \mathcal{R}_n), \mathcal{D}_n))$  be the partition of the hierarchical  $\Delta = (\text{Args}, \mathcal{R}, \mathcal{D})$ . Let  $AF_\Delta = (\text{Args}_\Delta, \mathcal{R}_\Delta)$ . Then  $AF_\Delta$  can be represented by the partition  $((\text{Args}'_1, \mathcal{R}'_1), \mathcal{R}'_{1-\mathcal{D}}), \dots, ((\text{Args}'_n, \mathcal{R}'_n), (\mathcal{R}'_{n-\mathcal{D}}))$  where:

- $\text{Args}_\Delta = \bigcup_{i=1}^n \text{Args}'_i$  and  $\mathcal{R}_\Delta = \bigcup_{i=1}^n (\mathcal{R}'_i \cup \mathcal{R}'_{i-\mathcal{D}})$
- for  $i = 1 \dots n$ ,  $(\text{Args}'_1, \mathcal{R}'_1)$  is an expansion of  $(\text{Args}_1, \mathcal{R}_1)$  and for  $i = 1 \dots n - 1$ ,  $(C, \overrightarrow{AB}) \in \mathcal{R}'_{i-\mathcal{D}}$  iff  $(C, (A, B)) \in \mathcal{D}_i$ , where  $C \in \text{Args}'_{i+1}$ ,  $\overrightarrow{AB} \in \text{Args}'_i$

**Lemma 2** Let  $\Delta = (\text{Args}, \mathcal{R}, \mathcal{D})$  be a hierarchical EAF.  $S$  is an admissible extension of  $\Delta$  iff  $T$  is an admissible extension of  $AF_\Delta = (\text{Args}_\Delta, \mathcal{R}_\Delta)$ , where:

$$T = S \cup \{\overrightarrow{XY} \mid Y \in S, X \rightarrow^S Y\} \cup \{\overrightarrow{YZ} \mid Y \in S, Y \rightarrow^S Z \text{ and there is a reinstatement set for } Y \rightarrow^S Z\}$$

**Proof** Let  $((\text{Args}_1, \mathcal{R}_1), \mathcal{D}_1), \dots, ((\text{Args}_n, \mathcal{R}_n), \mathcal{D}_n))$  be the partition of  $\Delta$ , and  $((\text{Args}'_1, \mathcal{R}'_1), \mathcal{R}'_{1-\mathcal{D}}), \dots, ((\text{Args}'_n, \mathcal{R}'_n), (\mathcal{R}'_{n-\mathcal{D}}))$  the partition of  $AF_\Delta$ . For  $i = 1 \dots n$ :

- 1)  $(\text{Args}_i, \mathcal{R}_i)$  and  $(\text{Args}'_i, \mathcal{R}'_i)$  are Dung argumentation frameworks, where  $(\text{Args}'_i, \mathcal{R}'_i)$  is an expansion of  $(\text{Args}_i, \mathcal{R}_i)$

2)  $\forall (X, Y) \in \mathcal{R}, \forall (Z, (X, Y)) \in \mathcal{D}, (X, Y) \in \mathcal{R}_i$  iff  $(Z, (X, Y)) \in \mathcal{D}_i, Z \in \text{Args}_{i+1}$

- 3) For  $i = 1 \dots n - 1$ ,  $(Z, W) \in \mathcal{R}'_{i-\mathcal{D}}$  implies  $Z \in \text{Args}'_{i+1}, W \in \text{Args}'_i$ , and  $W$  is an argument of the form  $\overrightarrow{XY}$ .

4) For  $i = 1 \dots n$ ,  $(Z, (X, Y)) \in \mathcal{D}_i$  iff  $(Z, \overrightarrow{XY}) \in \mathcal{R}'_{i-\mathcal{D}}$ .

1) - 3) imply that  $S$  can be partitioned into  $S_1 \cup \dots \cup S_n$ ,  $T$  into  $T_1 \cup \dots \cup T_n$ , and that the theorem is shown by proving by induction on  $i$ , the following result:

$S_i = S_i \cup \dots \cup S_n$  is admissible iff  $T_i = T_i \cup \dots \cup T_n$  is admissible, where:

$T_i = S_i \cup \{\overrightarrow{XY} \mid Y \in S_i, X \rightarrow^{S_i} Y\} \cup \{\overrightarrow{YZ} \mid Y \in S_i, Y \rightarrow^{S_i} Z \text{ and there is a reinstatement set for } Y \rightarrow^{S_i} Z\}$ .

*Base case (i = n):* Since  $\mathcal{D}_n = \emptyset$ ,  $\mathcal{R}'_{n-\mathcal{D}} = \emptyset$ , and  $(\text{Args}'_n, \mathcal{R}'_n)$  is the expansion of  $(\text{Args}_n, \mathcal{R}_n)$ ,

then the result is given by lemma 1, where trivially:

$\forall Y \in Sn, Tn: \forall X \text{ s.t. } (X, Y) \in \mathcal{R}_n, X \rightarrow^{Sn} Y, \forall Z \text{ s.t. } (Y, Z) \in \mathcal{R}_n, Y \rightarrow^{Sn} Z \text{ and there is a reinstatement set } \{Y \rightarrow^{Sn} Z\} \text{ for } Y \rightarrow^{Sn} Z.$

*Inductive hypothesis (IH):* The result holds for  $j > i$ .

*General case:*

*Left to Right half:* Let  $A \in S_i$ . Suppose some  $B \in S_i$  s.t.  $B \rightarrow^{Si} A$ , based on the attack  $(B, A) \in \mathcal{R}_i$ . Since  $A$  acceptable w.r.t.  $Si$ ,  $\exists C \in S_i$ , s.t.  $C \rightarrow^{Si} B$ , based on  $(C, B) \in \mathcal{R}_i$ . By definition of  $Ti$ , and given 1), 3) and lemma 1,  $A, \overrightarrow{CB}$  and  $C$  are all in  $Ti$  and are acceptable w.r.t.  $Ti$  if we can show that  $\overrightarrow{CB}$  is acceptable w.r.t.  $Ti$  given some  $D \in T_{i+1}, (D, \overrightarrow{CB}) \in \mathcal{R}'_{i-D}$ .

By 4),  $(D, (C, B)) \in \mathcal{D}_i$ . By assumption of  $A$  is acceptable w.r.t.  $Si$ , then by 2),  $\exists E \in S_{i+1}$  s.t.  $E \rightarrow^{Si} D$  based on the attack  $(E, D) \in \mathcal{R}_{i+1}$ , and there is a reinstatement set for  $E \rightarrow^{Si} D$ . By IH,  $E \in T_{i+1}, \overrightarrow{ED} \in T_{i+1}$ , and since  $(\overrightarrow{ED}, D) \in \mathcal{R}'_{i+1}$ ,  $\overrightarrow{CB}$  is acceptable w.r.t.  $Ti$ .

It remains to show that for  $A \in S_i, A \in Ti$ , for any  $X$  such that  $A \rightarrow^{Si} X$ , and there is a reinstatement set for  $A \rightarrow^{Si} X$ , then  $\overrightarrow{AX} \in S_i$ . Since  $A \rightarrow^{Si} X, (A, X) \in \mathcal{R}_i$ , then  $\{(A, \overrightarrow{A}), (\overrightarrow{A}, \overrightarrow{AX}), (\overrightarrow{AX}, X)\} \subseteq \mathcal{R}'_i$ . Since  $A \rightarrow^{Si} X, \neg \exists Z \in S_{i+1}$  s.t.  $(Z, (A, X)) \in \mathcal{D}(\mathcal{D}_i)$ . By IH,  $\neg \exists Z \in T_{i+1}$  s.t.  $(Z, \overrightarrow{AX}) \in \mathcal{R}'_{i-D}$ . Hence,  $\overrightarrow{AX} \in Ti$  is acceptable w.r.t.  $Ti$  as it is reinstated from  $\overrightarrow{A}$ 's attack by  $A \in Ti$ .

*Right to Left half:* Let  $A \in Ti$  for some  $A \in Args$ . We show that  $A$  is acceptable w.r.t.  $Si$ . Suppose some  $\overrightarrow{BA}$  such that  $(\overrightarrow{BA}, A) \in \mathcal{R}'_i$ . Hence, by definition of  $Ti$ , and given 1), lemma 1 shows that:

1.  $\overrightarrow{B} \in Ti$ , and if  $(X, \overrightarrow{B}) \in \mathcal{R}'_i$  then  $X = B$ . Hence  $(B, A) \in \mathcal{R}$ . Assume  $\neg \exists X \in T_{i+1}$  s.t.  $(X, \overrightarrow{BA}) \in \mathcal{R}'_{i-D}$ . By IH and 4),  $\neg \exists X \in S_{i+1}, (X, (B, A)) \in \mathcal{D}_i$ , and so given 2),  $B \rightarrow^{Si} A$ .
2.  $\exists \overrightarrow{CB} \in Ti$  s.t.  $(\overrightarrow{CB}, B) \in \mathcal{R}'_i$ .  $(\overrightarrow{C}, \overrightarrow{CB}) \in \mathcal{R}'_i, (C, \overrightarrow{C}) \in \mathcal{R}'_i$ , where  $C \in Args, C \in S_i$ . Since  $Ti$  is conflict free,  $\neg \exists X \in T_{i+1}$  s.t.  $(X, \overrightarrow{CA}) \in \mathcal{R}'_{i-D}$ . By IH and 4),  $\neg \exists X \in S_{i+1}, (X, (C, B)) \in \mathcal{D}_i$ , and so given 2),  $C \rightarrow^{Si} B$ .

Suppose some  $X \notin T_{i+1}, (X, \overrightarrow{CA}) \in \mathcal{R}'_{i-D}$ . Given 4),  $(X, C, A) \in \mathcal{D}_i$ . By assumption of  $\overrightarrow{CA}$  acceptable w.r.t.  $Ti$ ,  $\exists \overrightarrow{YX} \in T_{i+1}, (\overrightarrow{YX}, X) \in \mathcal{R}'_{i+1}$ . By IH and definition of  $Ti$ ,  $Y \in T_{i+1}, Y \in S_{i+1}, Y \rightarrow^{Si+1} X, Y$  is acceptable w.r.t.  $Si + 1$ , and there is a reinstatement set  $R_{Si+1}$  for  $Y \rightarrow^{Si+1} X$ . Hence, there is a reinstatement set  $R_{Si} = R_{Si+1} \cup \{C \rightarrow^{Si} B\}$  for  $C \rightarrow^{Si} B$ . Hence,  $A$  is acceptable w.r.t.  $Si$ .

## References

- [1] T.J.M. Bench-Capon. Agreeing to Differ: Modelling Persuasive Dialogue Between Parties Without a Consensus About Values, *Informal Logic*, 22(3), 231-45, 2002.
- [2] T.J.M. Bench-Capon. Persuasion in Practical Argument Using Value-based Argumentation Frameworks, *Journal of Logic and Computation*, 13(3), 429-448, 2003.
- [3] T.J.M. Bench-Capon, S.Doutre and P.E. Dunne. Audiences in Argumentation Frameworks, *Artificial Intelligence*, 171, 42-71, 2007.
- [4] P.E. Dunne and T.J.M. Bench-Capon. Two party immediate response disputes: Properties and efficiency, *Artificial Intelligence*, 149, 221-250, 2002.
- [5] P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games, *Artificial Intelligence*, 77:321-357, 1995.
- [6] S. Modgil, *Value Based Argumentation in Hierarchical Argumentation Frameworks*. In: Proc. 1st Int. Conference on Computational Models of Argument, 297-308, Liverpool, UK, 2006.
- [7] S.Modgil. *An Abstract Theory of Argumentation That Accommodates Defeasible Reasoning About Preferences*. In: Proc. 9th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 648-659, 2007.
- [8] S. Modgil. *Reasoning About Preferences in Argumentation Frameworks*. Techincal Report: <http://www.dcs.kcl.ac.uk/staff/modgilsa/ArguingAboutPreferences.pdf>
- [9] J.R. Searle. Rationality in Action, MIT Press, Cambridge, MA, 2001.

# Applying Preferences to Dialogue Graphs

Sanjay Modgil<sup>a1</sup> Henry Prakken<sup>b</sup>

<sup>a</sup> Department of Computer Science, Kings College London

<sup>b</sup> Department of Information and Computing Sciences, University of Utrecht and  
Faculty of Law, University of Groningen

**Abstract.** An abstract framework for formalising persuasion dialogues has recently been proposed. The framework provides for a range of speech acts, and protocols of varying levels of flexibility. However, the framework assumes the availability of preference information relevant to determining whether arguments moved in a dialogue *defeat* each other. However, preference information may only become available after the dialogue has terminated. Hence, in this paper, we describe dialogues conducted under the assumption of an *attack* relation that does not account for preferences. We then describe how the resultant dialogue graph can be pruned by a preference relation in order to determine whether the winner of the dialogue is still the winner given the newly available preference information. We also describe a class of protocols that account for subsequent pruning by a preference relation, and show that under a restriction on the pruning, if the player defending the dialogue's main topic is winning the dialogue, then (s)he remains the winner irrespective of the preference relation applied.

## 1. Introduction

In *persuasion dialogues*, participants attempt to persuade other participants to adopt their point of view. Argumentation based formalisations of such dialogues (reviewed in [3]) generally adopt a game theoretic approach in which speech acts are viewed as moves in a game that are regulated by the game's rules (protocol). A recent formal framework for argumentation based dialogue games [7] abstracts from the specific speech acts, requiring only that they conform to an explicit reply structure; each dialogue move attacks or surrenders some earlier move of the other participant. The current winner of a dialogue can then be determined at any stage, based on the *dialogical status* of moves, the evaluation of which exploits the explicit reply structure. Furthermore, the dialogical status of moves also provides for maintaining the focus of a dialogue when encoding protocols of varying levels of flexibility (*single* and *multi-move*, and *single* and *multi-reply* protocols). The framework's level of abstraction and dialectical nature thus mirrors that of a Dung argumentation framework [4] in which arguments are related by a binary conflict based relation of *attack* or *defeat*. It allows for different underlying logics, and speech acts that can be modelled as nodes in a dialogue graph, whose dialogical status is evaluated based on the reply relations.

---

<sup>1</sup>This author is supported by the EU 6th Framework project CONTRACT (INFSO-IST-034418). The opinions expressed herein are those of the named authors only and should not be taken as necessarily representative of the opinion of the European Commission or CONTRACT project partners.

For example, consider the initial move of a proponent  $P$  claiming:  $P_1$  = “My Mercedes is safe”. An opponent  $O$  may then query with an attacking reply on  $P_1$ :  $O_2$  = “Why is your Mercedes safe”. The idea is that  $O$  is the current winner, as the burden of proof is on  $P$  to reply with an argument for its claim. The *dialogical status* of  $P_1$  is said to be *out*, and  $O_2$  is said to be *in*.  $P$  may surrender to  $O_2$  with a reply that retracts the claim  $P_1$ , or may in turn attack  $O_2$  with the required argument:  $P_3$  = “My Mercedes is safe since it has an airbag”.  $O_2$  is now *out*,  $P_3$  and  $P_1$  *in*, and  $P$  is the current winner.  $O$  might move a surrendering reply to  $P_3$ , conceding the argument’s claim, or move an attacking reply to  $P_3$ , counter-arguing that Mercedes cars are not safe.

The framework of [7] assumes that if an argument  $B$  replies to another argument  $A$ , then it is required that  $B$  *defeats*  $A$ , assuming the standard:

*if  $B$  attacks  $A$ , then  $B$  defeats  $A$  only if  $A$  is not stronger than (preferred to)  $B$ .*

However, it may be that a preference relation on arguments can only be applied **after** termination of the dialogue. Hence,  $B$  may reply to  $A$ , but only after the dialogue is it determined that  $A$  is in fact stronger than  $B$ ; information that would have invalidated moving  $B$  had it been known at the time of the dialogue. For example in legal adjudication procedures [8] an adjudicator typically decides the dispute after the dialogue between the adversaries has been terminated, so that in that dialogue the preferences that the adjudicator will apply are not yet known. Also, consider the CARREL system [10] developed to demonstrate the ASPIC project’s argumentation components ([www.argumentation.org](http://www.argumentation.org)). CARREL facilitates argumentation between heterogeneous agents (software and human) over the validity of organs for transplantation. The dialogues are mediated by a component implementing a protocol in the framework of [7]. Currently, the strengths of arguments are assumed to be given at the time of the dialogue. However, a fully realised CARREL system requires that arguments’ strengths be determined after the dialogue. In [11], development of a case based reasoning engine is described. A constructed dialogue graph is input to an algorithm that determines the relative strengths of (and so defeats between) arguments that symmetrically attack. This is based on a comparison with dialogue graphs associated with previous cases, where the same arguments were used, and where the success of the ensuing transplant is then used to weight these arguments in the new graph.

In section 2 of this paper we review the framework of [7] and introduce some new concepts required for section 3. In section 3 we describe dialogues conducted under the assumption that arguments *attack* rather than *defeat* their target arguments. After termination of a dialogue, a preference relation on the arguments moved in a dialogue is then applied to the dialogue graph, removing argue moves (and the moves that follow these) that would have been invalidated had the preference information been available at the time of the dialogue. We then describe protocols that accounts for *attacks* rather than *defeats*. We then show a result of theoretical and practical interest that holds under certain conditions; viz. if the initial move is *in* in a dialogue, then it remains *in* irrespective of the preference relation applied to the arguments. Finally section 4 concludes with a discussion of future work.

## 2. A General Framework for Persuasion Dialogues

In the framework of [7], dialogues are assumed to be for two parties; the *proponent* ( $P$ ) who defends the dialogue topic  $t$  and the *opponent* ( $O$ ) who challenges  $t$ .

**Table 1.**  $L_c$  for liberal dialogues

Locutions	Attacks	Surrenders
<i>claim</i> $\varphi$	<i>why</i> $\varphi$	<i>concede</i> $\varphi$
<i>why</i> $\varphi$	<i>argue A</i> ( $\text{conc}(A) = \varphi$ )	<i>retract</i> $\varphi$
<i>argue A</i>	<i>why</i> $\varphi$ ( $\varphi \in \text{prem}(A)$ ) <i>argue B</i> ( $(B, A) \in \mathcal{R}$ )	<i>concede</i> $\varphi$ ( $\varphi \in \text{prem}(A)$ or $\varphi = \text{conc}(A)$ )
<i>concede</i> $\varphi$		
<i>retract</i> $\varphi$		

**Definition 1** A *dialogue system for argumentation* (dialogue system for short) is a pair  $(\mathcal{L}, \mathcal{D})$ , where  $\mathcal{L}$  is a logic for defeasible argumentation, and  $\mathcal{D}$  is a triple  $(L_c, \mathbb{P}, C)$  where  $L_c$  is a communication language,  $\mathbb{P}$  a *protocol* for  $L_c$ , and  $C$  specifies the effects of locutions in  $L_c$  on the participants' *commitments*.

**Definition 2** A *logic for defeasible argumentation*  $\mathcal{L}$  is a tuple  $(L_t, \text{Inf}, \text{Args}, \mathcal{R})$ , where  $L_t$  (the *topic language*) is a logical language,  $\text{Inf}$  is a set of *inference rules* over  $L_t$ ,  $\text{Args}$  (the *arguments*) is a set of AND-trees of which the nodes are in  $L_t$  and the AND-links are inferences instantiating rules in  $\text{Inf}$ , and  $\mathcal{R}$  is a binary relation on  $\text{Args}$ . For any argument  $A$ ,  $\text{prem}(A)$  is the set of leaves of  $A$  (its premises) and  $\text{conc}(A)$  is the root of  $A$  (its conclusion).

An argument  $B$  *backward extends* an argument  $A$  if  $\text{conc}(B) = \phi$  and  $\phi \in \text{prem}(A)$ . The concatenation of  $A$  and  $B$  (where  $B$  *backward extends*  $A$ ) is denoted by  $B \otimes A$ .

Defeasible inference in  $\mathcal{L}$  is assumed to be defined according to the grounded semantics [4]. Note that the framework abstracts from the nature of the rules in  $\text{Inf}$ . In this paper, we assume  $\mathcal{R}$  denotes either a conflict based *attack* relation, or a *defeat* relation that additionally accounts for preference information. From hereon, we write  $A \rightarrow B$  to denote that  $(A, B) \in \mathcal{R}$ , and  $A \leftrightarrow B$  to denote  $(A, B), (B, A) \in \mathcal{R}$ . In this paper we will ignore commitment rules since they are not relevant to the work presented here. A communication language is a set of locutions and two relations of attacking and surrendering reply defined on this set.

**Definition 3** A *communication language* is a tuple  $L_c = (S, R_a, R_s)$ , where:

$S$  is a set of *locutions* such that each  $s \in S$  is of the form  $p(c)$ , where  $p$  is an element of a given set of performatives, and  $c$  either is a member or subset of  $L_t$ , or is a member of  $\text{Args}$  (of some given logic  $\mathcal{L}$ ).

$R_a$  and  $R_s$  are irreflexive *attacking* and *surrendering reply* relations on  $S$  that satisfy:

1.  $\forall a, b, c : (a, b) \in R_a \Rightarrow (a, c) \notin R_s$  (a locution cannot be an attack and a surrender at the same time)
2.  $\forall a, b, c : (a, b) \in R_s \Rightarrow (c, a) \notin R_a$  (surrenders cannot be attacked since they effectively end a line of dispute)

An example  $L_c$  is shown in Table 1 (in which we write ' $(A, B) \in \mathcal{R}$ ' rather than ' $A$  defeats  $B$ ' as in [7]). For each locution, its surrendering replies and attacking replies are shown in the same row, where the latter are said to be the *attacking counterparts* of the row's surrendering replies. Note however, that for the second line of the *argue A* row, *argue B* is an attacking counterpart of *concede*  $\varphi$  only if the conclusion of  $B$  negates or is negated by  $\varphi$ . (So the attacking counterpart of conceding a premise is a premise-attack and the attacking counterpart of conceding a conclusion is a rebuttal.).

In general, the protocol for a communication language  $L_c$  is defined in terms of the notion of a dialogue, which in turn is defined with the notion of a move:

**Definition 4** Let  $L_c = (S, R_a, R_s)$ . The set  $M$  of moves is defined as  $\mathbb{N} \times \{P, O\} \times S \times \mathbb{N}$ , where the four elements of a move  $m$  are denoted by, respectively:

- $id(m)$ , the *identifier* of the move,
- $pl(m)$ , the *player* of the move,
- $s(m)$ , the *locution* performed in the move,
- $t(m)$ , the *target* of the move.

The set of (finite) dialogues, denoted by  $(M^{<\infty}) M^{\leq\infty}$ , is the set of all (finite) sequences  $m_1, \dots, m_i, \dots$  from  $M$  such that: each  $i^{th}$  element in the sequence has identifier  $i$ ;  $t(m_1) = 0$ ; for all  $i > 1$  it holds that  $t(m_i) = j$  for some  $m_j$  preceding  $m_i$  in the sequence.

For any dialogue  $d = m_1, \dots, m_n, \dots$ , the sequence  $m_1, \dots, m_i$  is denoted by  $d_i$ , where  $d_0$  denotes the empty dialogue. When  $d$  is a dialogue and  $m$  a move then  $d, m$  denotes the continuation of  $d$  with  $m$ .

In general we say ‘ $m$  is in  $d$ ’ if  $m$  is a move in the sequence  $d = m_1, \dots, m_i, \dots$ . When  $t(m) = id(m')$  we say  $m$  replies to its target  $m'$  in  $d$ , and abusing notation we may let  $t(m)$  denote a move instead of just its identifier. When  $s(m)$  is an attacking (surrendering) reply to  $s(m')$  we also say  $m$  is an attacking (surrendering) reply to  $m'$ .

We now review [7]’s protocol rules that capture a lower bound on the coherence of dialogues. Note that since each move in a dialogue is a reply to a single earlier move, then any dialogue can be represented as a tree. Prior to presenting the protocol rules we define in this paper the notion of a line in a dialogue, and a representation of a finite dialogue as the set of paths (from root node to leaf) or ‘lines’ (of dialogue) that constitute the dialogue tree whose root node is the initial move  $m_1$ .

**Definition 5** Let  $d$  be a finite dialogue with initial move  $m_1$ , and let  $leaves(d) = \{m \mid m$  is a move in  $d$ , and no  $m'$  in  $d$  replies to  $m\}$ .

- Let  $m_k$  be any move in  $d$ . Then  $line(d, m_k) = m_1, \dots, m_k$ , where for  $j = 2 \dots k$ ,  $t(m_j) = m_{j-1}$ .
- Let  $leaves(d) = \{m_{k_1}, \dots, m_{k_n}\}$ . Then  $lines(d) = \bigcup_{i=k_1}^{k_n} line(d, m_{k_i})$ .
- Let  $l = m_1, \dots, m_k \in lines(d)$ . Then, for  $i = 1 \dots k$ ,  $l' = m_1, \dots, m_i$  is a sub-line of  $l$ .
- If  $l = m_1, \dots, m_i$ ,  $l' = m_1, \dots, m_i, \dots, m_j$ , then  $l'$  is said to be larger than  $l$ .

**Definition 6** A *protocol* on  $M$  is a set  $\mathbb{P} \subseteq M^{<\infty}$  such that whenever  $d$  is in  $\mathbb{P}$ , so are all initial sequences that  $d$  starts with. A partial function  $Pr : M^{<\infty} \rightarrow \mathcal{P}(M)$  is derived from  $\mathbb{P}$  as follows:

- $Pr(d) = \text{undefined whenever } d \notin \mathbb{P}$ ;
- $Pr(d) = \{m \mid d, m \in \mathbb{P}\}$  otherwise.

The elements of  $dom(Pr)$  (the domain of  $Pr$ ) are called the *legal finite dialogues*. If  $d$  is a legal dialogue and  $Pr(d) = \emptyset$ , then  $d$  is said to be a *terminated* dialogue.

Let  $T$  be a turntaking function  $T : M^{<\infty} \rightarrow \mathcal{P}(\{P, O\})$  such that  $T(\emptyset) = \{P\}$ .

Then  $(\mathcal{L} = (L_t, Inf, Args, \mathcal{R}), \mathcal{D} = (L_c, \mathbb{P}, C))$  is a coherent dialogue system if  $\mathbb{P}$  satisfies the following basic conditions for all moves  $m$  and all legal finite dialogues  $d$ .

If  $m \in Pr(d)$ , then:

- $R_1$ :  $pl(m) \in T(d)$ ;
- $R_2$ : If  $d \neq d_0$  and  $m \neq m_1$ , then  $s(m)$  is an attacking or surrendering reply to  $s(t(m))$  according to  $L_c$ ;
- $R_3$ : If  $m$  replies to  $m'$ , then  $pl(m) \neq pl(m')$ ;
- $R_4$ : If there is an  $m'$  in  $d$  such that  $t(m) = t(m')$  then  $s(m) \neq s(m')$ .
- $R_5$ : For any  $m' \in d$  that surrenders to  $t(m)$ ,  $m$  is not an attacking counterpart of  $m'$  (no move has both a surrender and its attacking counterpart).
- $R_6$ : For any  $m' \in d$  such that  $pl(m') = P$ ,  $s(m') = \text{argue } B$ ,  $m'$  replies to  $m$ ,  $s(m) = \text{argue } A$ , then  $\text{argue } B$  is not in  $\text{line}(d, m)$ .

Note that  $R_2$  combined with Table 1's requirement that an *argue* reply must be in the relation  $\mathcal{R}$  to its target, effectively builds a version of the argument-game proof theory of [9] into the protocol. Note also that in this paper we have added  $R_6$  to the basic rules of [7] to avoid unnecessary non-termination of dialogues. It is known that this rule (but not the corresponding rule for  $O$ ) does not change soundness and completeness of the argument game with respect to grounded semantics.

So far any ‘verbal struggle’ can fit the above framework. It can be specialised for a particular communication language and associated set of protocol rules. Here, we review [7]’s class of *liberal* dialogue systems (parameterised by a logic  $\mathcal{L}$ ) that make use of  $L_c$  in Table 1, and in which the participants have much freedom. Two additional protocol rules are added to those in definition 6:

$R_7$  : If  $d = \emptyset$  then  $s(m)$  is of the form  $\text{claim}(\phi)$  or  $\text{argue } A$  (proponent  $P$  starts with a unique move that is a claim or argument)

$R_8$  : if  $m$  concedes the conclusion of an argument in  $m'$ , then  $m'$  does not reply to a *why* move (ensuring that only conclusions of counter-arguments can be conceded)

Consider the following dialogue (where  $\phi$  since  $\alpha_1, \dots, \alpha_n$  denotes an argument):

**Example 1** [Example Dialogue]

$P_1: a$ since $g, f$	
$O_2: concede f$	( $O_2$ is a surrendering reply to $P_1$ )
$O_3: why g$	( $O_3$ is an attacking reply to $P_1$ )
$P_4: g$ since $\neg b$	( $P_4$ is an attacking reply to $O_3$ )
$O_5: \neg g$ since $\neg g$	( $O_5$ attack replies to $P_4$ with the fact $\neg g$ )

As discussed in section 1, [7] defines evaluation of the current winner of a dialogue based on the *dialogical status* of moves.

**Definition 7** Let a move  $m$  in a dialogue  $d$  be *surrendered* iff it is an *argue A* move and it has a reply in  $d$  that concedes  $A$ ’s conclusion; or else  $m$  has a surrendering reply in  $d$ . Then, all attacking moves in a finite dialogue  $d$  are either *in* or *out* in  $d$ . Such a move  $m$  is *in* iff

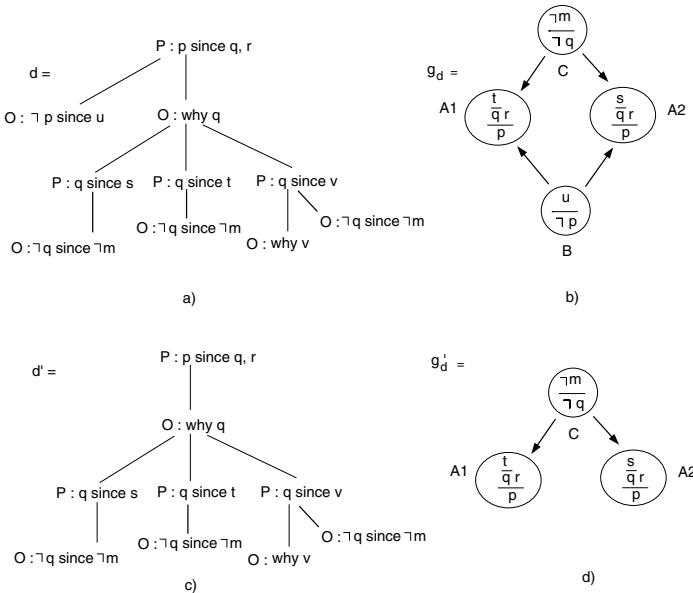
1.  $m$  is surrendered in  $d$ ; or else
2. all attacking replies to  $m$  are *out*

Otherwise  $m$  is *out*.

We say that if the status of the initial move  $m_1$  in *in* (*out*) then  $P$  ( $O$ ) currently wins  $d$ .

In example 1,  $O_2$  is a surrendering reply to  $P_1$ , but  $P_1$  is not surrendered since its conclusion has not been conceded.  $O$  is the current winner since  $O_5$  is *in*, hence its target  $P_4$  is made *out*, hence  $O_3$  is *in*, and so  $P_1$  is *out*.

During the course of a dialogue  $d$ , players implicitly build a dialectical graph  $g_d = (\text{Args}_d, \mathcal{R}_d)$  (a Dung graph [4]) of arguments and counter-arguments related to the dialogue topic (see [7] for details of how  $d_g$  is constructed). Intuitively,  $\text{Args}_d$  contains any  $A$  in an argue move, backward extended on premises that are not challenged (by *why* locutions) or retracted, provided  $A$  does not itself backward extend an argument.  $\mathcal{R}_d$  is then defined by the reply relations in the graph. For example, consider the dialogue in figure 1a, and its associated dialectical graph in figure 1b, consisting of two arguments for  $p$  constructed during the dialogue.



**Figure 1.** Dialogues and their dialectical graphs. Note that  $(p|q, r|v)$  is not in b) and d) since  $v$  is under challenge by a *why* locution.

### 3. Applying Preferences to Dialogues

#### 3.1. Preference based Resolution of Dialogue Graphs

Thus far we have assumed that if a move *argue*  $B$  replies to *argue*  $A$ , then  $(B, A) \in \mathcal{R}$  in the underlying logic, where  $\mathcal{R}$  denotes either an *attack* or *defeat* relation. As discussed in section 1, scenarios where a preference relation on arguments (based on their relative strengths) is available only after the dialogue, require a focus on dialogues where  $\mathcal{R}$  denotes *attack*. As mentioned earlier, the framework of [7] assumes that backwards extending of arguments does not weaken arguments. This assumption is made explicit here.

**Definition 8** Let  $d$  be a dialogue in the dialogue system  $(\mathcal{L} = (L_t, Inf, Args, \mathcal{R}), \mathcal{D})$ . We say that a preordering  $Pref$  on  $Args$  is a *preference relation* on  $Args$ , where:

$\gg^{Pref}$  is the associated strict ordering ( $A \gg^{Pref} B$  iff  $APrefB$  and not  $BPrefA$ )

$\equiv^{Pref}$  is the associated relation of equivalence ( $A \equiv^{Pref} B$  iff  $APrefB$  and  $BPrefA$ )

We say that  $Pref$  is *not weakened by backward extending* iff:

$\forall A, B \in Args$ , if  $A \gg^{Pref} B$ , then  $\forall A', B'$  s.t.  $A' \otimes A, B' \otimes B \in Args$ ,  $A' \otimes A \gg^{Pref} B' \otimes B$

From hereon we will, unless stated otherwise, assume that  $Pref$  is *not weakened by backward extending*. Notice that the backward extension restriction is satisfied by the last link valuation of argument strength (used in [9] and the CARREL [10] system described in section 1). In figure 1a), suppose  $(p \text{ since } q, r) \gg^{Pref} (\neg p \text{ since } u)$ . If  $Pref$  is not weakened by backward extending, then  $A1 >^{Pref} B$  and  $A2 >^{Pref} B$  in the dialogue's dialectical graph.

We now define a resolution  $resolve(d, pn, Pref)$  of a dialogue  $d$  based on  $Pref$  and a pruning function  $pn$ , where  $pn$  takes as input a dialogue line  $l$  and a strict ordering  $\gg$ , and returns a sub-line of  $l$ . One can then determine the status of the initial move  $m_1$  in the resolution  $resolve(d, pn, Pref)$ , and thus determine whether, based on the available preference information, proponent remains as the winner or loser of the dialogue.

**Definition 9** Let  $d$  be a dialogue and  $lines(d) = \{l_1, \dots, l_n\}$ . Then:

$\bigcup_{i=1}^n pn(l_i, \gg^{Pref})$  is the resolution of  $d$  w.r.t.  $Pref$ , denoted  $resolve(d, pn, Pref)$ .

We now define a specific pruning function  $prune$ . Firstly, note that some argument game proof theories proposed for the grounded semantics (e.g. [1,9]) account for the extra burden of proof on  $P$ , by requiring that  $P$ 's arguments *strictly* defeat their targets (this requirement is not needed for soundness and completeness but makes dialogues shorter and therefore protocols more efficient). Recall that  $B$  defeats  $A$  if it attacks  $A$ , and  $A$  is not strictly preferred to  $B$ , and  $B$  *strictly* defeats  $A$  if  $B$  defeats  $A$  and  $A$  does not defeat  $B$ . Hence, if  $A$  and  $B$  symmetrically attack, then  $B$  *strictly* defeats  $A$  only if  $B \gg^{Pref} A$ . Hence, the pruning function will only retain a  $B$  moved by  $P$  as a reply to the symmetrically attacking  $A$ , if  $B \gg^{Pref} A$ .

**Definition 10** Let  $d$  be a dialogue in  $(\mathcal{L} = (L_t, Inf, Args, \mathcal{R}), \mathcal{D})$ . Let  $l \in lines(d)$  and  $\gg$  a strict ordering on the arguments moved in  $l$ . Then  $l' = prune(l, \gg)$  is the largest sub-line of  $l$  such that for all  $m'$  in  $l'$ , if  $m'$  replies to  $m$ , and  $s(m) = argue A$ ,  $s(m') = argue B$ , then:

1. it is not the case that  $A \gg B$  (it is not the case that  $B$  does not defeat  $A$  according to  $\gg$ )
2. if  $pl(m') = P$ ,  $pl(m) = O$ , and  $(A, B) \in \mathcal{R}$ , then  $B \gg A$  ( $B$  strictly defeats  $A$  according to  $\gg$ )

**Example 2** Consider dialogue  $d$  in figure 1a). The initial *argue* move  $p \text{ since } q, r$  is *out*. Suppose  $(p \text{ since } q, r) \gg^{Pref} (\neg p \text{ since } u)$ , and  $(\neg q \text{ since } \neg m) \equiv^{Pref} (q \text{ since } s)$ .  $d' = resolve(d, prune, Pref)$  is shown in figure 1c) in which *argue* move  $p \text{ since } q, r$  remains *out*.  $d'$ 's dialectical graph  $g'_d$  is shown in figure 1d). Note that since  $Pref$  is not weakened by backward extending, then  $A1 \gg^{Pref} B$  and  $A2 \gg^{Pref} B$ , suggesting that  $g'_d$  might be directly obtained by applying  $Pref$  to  $g_d$  (although this is not trivial and remains a topic for future work).

Finally, note that in [7] it is shown that for dialogues without surrenders in which the players play logically perfectly (the players move any attacking arguments defined by the dialogue's dialectical graph, as for example shown in figure 1a)), the initial move is *in* iff the main argument is in the grounded extension of the argument's dialectical graph. It immediately follows from the proofs in [7] that this result still holds for resolved dialogue graphs that satisfy the same properties.

### 3.2. 'Rational' Protocols

Consider a liberal dialogue  $d$  where  $A, B$  and  $C$  are the arguments moved, and assume  $C \leftrightarrow B \rightarrow A$ . Suppose the dialogue starts with:

$$d_2 = P_1 : \text{argue } A, O_2 : \text{argue } B.$$

As previously observed, argument games for the grounded semantics would prohibit  $P_3 : \text{argue } C$  (given that  $B \rightarrow C$ ). As required,  $P_1$  is *out* since  $A$  is not in the grounded extension. However, assuming  $\rightarrow$  denotes an attack relation, and subsequent application of preferences, then a 'rational' proponent would move  $P_3 : \text{argue } C$ , so that if it turns out that  $C \gg^{\text{Pref}} B$ , then  $P_1$  is now *in* in the dialogue:

$$d_3 = P_1 : \text{argue } A, O_2 : \text{argue } B, P_3 : \text{argue } C.$$

However, note now that proponent is the current winner of the dialogue, despite the fact that  $A$  is not in the grounded extension of  $C \leftrightarrow B \rightarrow A$ . A rational opponent would move  $B$  again. We therefore state that an opponent plays a dialogue *rationally* if the dialogue satisfies the following:

**Definition 11** [Rational Opponent]  $R_{RO}$ : If  $d$  contains a subsequence of moves  $\dots, m_j, \dots, m_k, \dots$  such that  $m_k$  replies to  $m_j$ , and  $m_j = O : \text{argue } A, m_k = P : \text{argue } B$ , then: if  $(A, B) \in \mathcal{R}$ , and there is no  $m_l$  in  $d$  that replies to  $m_k$  such that  $m_l = O : \text{argue } A$ , then  $m = O : \text{argue } A$  replies to  $m_k$ .

Hence, *argue A* is made *out* in:

$$d_4 = P_1 : \text{argue } A, O_2 : \text{argue } B, P_3 : \text{argue } C, O_4 : \text{argue } B,$$

Suppose  $d_4$  continues with proponent's attacking reply querying a premise  $\phi$  in the second instance of  $B$ , obtaining:

$$d_5 = P_1 : \text{argue } A, O_2 : \text{argue } B, P_3 : \text{argue } C, O_4 : \text{argue } B, P_5 : \text{why } \phi.$$

Observe now that if  $B \gg^{\text{Pref}} C$ , then  $\text{resolve}(d_5, \text{prune}, \text{Pref}) = P_1 : \text{argue } A, O_2 : \text{argue } B$  in which  $A$  is *out*! Proponent's attacking reply on  $B$  has been 'lost' in the pruning. A rational proponent would therefore challenge  $\phi$  in the first instance of  $B$ . We therefore also state that a proponent plays a dialogue *rationally* if it always replies to the first occurrence in a dialogue line of an argue move by opponent. It is easy to see that this will never deny proponent any opportunity, since any reply that is effective on a later occurrence will be effective on the first occurrence.

**Definition 12** [Rational Proponent]  $R_{RP}$ : If  $m_k$  is a move  $O : \text{argue } A$  in a dialogue  $d$ , and  $m$  replies to  $m_k$ , then letting  $\text{line}(d, m_k) = m_1, \dots, m_k$ , for  $i < k$ ,  $m_i \neq O : \text{argue } A$ .

Liberal dialogues in which opponent and proponent play rationally are from hereon referred to as *rational liberal dialogues*.

Intuitively, we would want that the resolution  $d'$  of a rational liberal dialogue  $d$  by  $\text{Pref}$ , is itself a dialogue that would have been obtained assuming availability of  $\text{Pref}$  and a defined defeat relation. However, the requirement that arguments moved by  $P$  strictly defeat their target arguments is not captured by protocol rules  $R_1 - R_8$ . We therefore refer to *grounded liberal* protocols that are additionally defined by the rule:

$$R_9 : \text{If } m = P : \text{argue } B \text{ replies to } m' = O : \text{argue } A, \text{ then } (A, B) \notin \mathcal{R}, (B, A) \in \mathcal{R}.$$

In the following proposition we say that line  $l$  is pruned at  $m_j$  to obtain  $l'$  ending in  $m_i$ , if  $l = m_1, \dots, m_i, m_j, \dots$  and  $l' = m_1, \dots, m_i$ .

**Proposition 1** Let  $d$  be a rational liberal dialogue in  $(\mathcal{L} = (L_t, \text{Inf}, \text{Args}, \mathcal{R}), \mathcal{D} = (L_c, \mathbb{P}, C))$  and let  $d' = \text{resolve}(d, \text{prune}, \text{Pref})$ .

Let  $\text{Pref}$  be a preordering on  $\text{Args}$  and  $\mathcal{R}' = \{(B, A) \mid (B, A) \in \mathcal{R} \text{ and it is not the case that } A \gg^{\text{Pref}} B\}$ . Then:

$d'$  is a dialogue in  $(\mathcal{L}' = (L_t, \text{Inf}, \text{Args}, \mathcal{R}'), \mathcal{D}' = (L_c, \mathbb{P}', C))$ , where  $\mathbb{P}'$  is a grounded liberal protocol, and:

for every  $l \in \text{lines}(d)$  pruned at  $m_j$  to obtain  $l'$  ending in  $m_i$ ,  $m_j$  is not a legal reply to  $m_i$  in  $\mathcal{D}'$ .

**Proof:** Let  $l = m_1, \dots, p : \text{argue } A, \bar{p} : \text{argue } B, \dots$  be any line in  $\text{lines}(d)$ , and suppose  $l$  is pruned to obtain  $l' = m_1, \dots, p : \text{argue } A$ . There are two cases to consider:

- 1)  $A \gg B$ , and i)  $(B, A) \in \mathcal{R}$ ,  $(A, B) \notin \mathcal{R}$ , or: ii)  $(B, A), (A, B) \in \mathcal{R}$ , and  $p = P, \bar{p} = O$ . Hence,  $(B, A) \notin \mathcal{R}'$  so that  $\text{argue } B$  is not a legal reply to  $\text{argue } A$  in the grounded liberal protocol.
- 2)  $p = O, \bar{p} = P, (B, A), (A, B) \in \mathcal{R}$ , and it is not the case that  $B \gg A$ . In which case  $(A, B) \in \mathcal{R}'$ .

In both cases (by definition of  $L_c$  (table 1) in the case of 1) and  $R_9$  in the case of 2))  $\bar{p}$  cannot move  $\text{argue } B$  as a reply to  $p : \text{argue } A$  in the grounded liberal game.

### 3.3. Applying Preferences Exclusively to Symmetric Attacks

We now go on to show a practically useful result that holds for rational liberal protocols. The result holds for argumentation formalisms in which preferences are applied only to symmetric attacks. Hence we give the following definition of  $\text{pruneS}$  ( $S$  for symmetric) that can be used in defining the resolution of a dialogue as described in definition 9.

**Definition 13** Let  $d$  be a dialogue in  $(\mathcal{L} = (L_t, \text{Inf}, \text{Args}, \mathcal{R}), \mathcal{D})$ ,  $l \in \text{lines}(d)$  and  $\gg$  a strict ordering on the arguments moved in  $l$ . Then  $l' = \text{pruneS}(l, \gg)$  is the largest sub-line of  $l$  such that for all  $m'$  in  $l'$ , if  $m'$  replies to  $m$ , and  $s(m) = \text{argue } A, s(m') = \text{argue } B$ , **and**  $(A, B), (B, A) \in \mathcal{R}$ , then:

1. it is not the case that  $A \gg B$
2. if  $p(m') = P, p(m) = O$ , and  $(A, B) \in \mathcal{R}$ , then  $B \gg A$

Proposition 2 below states that under the assumption of rational liberal protocols, and application of preferences to symmetric attacks, then if the status of the initial move  $m_1$  is *in*, then it is **in irrespective** of the preference ordering applied. Prior to showing the proposition we define the *line status* of a move and establish a lemma (notice that surrendering replies only terminate lines since they cannot be replied to).

**Definition 14** Let  $l = m_1, \dots, m_n$  be a line in a finite dialogue  $d$  ( $l \in \text{lines}(d)$ ).

- If  $m_n$  is a surrendering reply to  $m_{n-1}$ , then  $\text{line-status}(l, m_{n-1})$  is *in*, and for  $i < n-1$ :  $\text{line-status}(l, m_i)$  is *in* if  $\text{line-status}(l, m_{i+1})$  is *out*, else  $\text{line-status}(l, m_i)$  is *out*.
- If  $m_n$  is an attacking reply to  $m_{n-1}$ , then  $\text{line-status}(l, m_n)$  is *in*, and for  $i < n$ :  $\text{line-status}(l, m_i)$  is *in* if  $\text{line-status}(l, m_{i+1})$  is *out*, else  $\text{line-status}(l, m_i)$  is *out*.

**Lemma 1** Let the dialogical status of the initial move  $m_1$  in dialogue  $d$  be *in*. Let  $\text{lines}(d) = \{l_1, \dots, l_n\}$ , and  $\text{resolve}(d, pn, \text{Pref}) = \{l'_1, \dots, l'_n\}$ . Then, if for  $i = 1, \dots, n$ , if for all  $m$  in  $l'_i$  such that  $m$  is in  $l$ , either:

- $\text{line-status}(l'_i, m) = \text{line-status}(l_i, m)$ ; or
- if  $\text{line-status}(l'_i, m) \neq \text{line-status}(l_i, m)$ , then either
  - \*  $\text{line-status}(l'_i, m) = \text{in}$ , and  $p(m) = P$ , or
  - \*  $\text{line-status}(l'_i, m) = \text{out}$  and  $p(m) = O$

then the dialogical status of the initial move  $m_1$  in the dialogue  $d'$  corresponding to  $\{l'_1, \dots, l'_n\}$  is *in*.

**Proof:** Obvious.

**Proposition 2** Let  $d = m_1, \dots, m_n$  be a rational liberal finite dialogue in  $(\mathcal{L} = (L_t, Inf, Args, \mathcal{R}), \mathcal{D})$ . If the dialogical status of  $m_1$  is *in*, then for all preorderings  $\text{Pref}$  on  $Args$ ,  $m_1$  is *in* in  $\text{resolve}(d, \text{pruneS}, \text{Pref})$ .

**Proof:** Suppose some line  $l \in \text{lines}(d)$  and arguments  $A, B$  such that  $(A, B), (B, A) \in \mathcal{R}$  and either  $A \gg^{\text{Pref}} B$ , or it is not the case that  $B \gg^{\text{Pref}} A$ . There are three cases where  $l$  is pruned to obtain  $l'$  (we simply write the players and the arguments):

1.  $m_1, \dots, P_i : A, O_{i+1} : B, \dots$
2.  $m_1, \dots, O_i : A, P_{i+1} : B, O_{i+2} : A$  (by  $R_{RO}$ )
3.  $m_1, \dots, O_i : B, P_{i+1} : A, O_{i+2} : B$  (by  $R_{RO}$ )

(Note that  $\dots, P_i : B, O_{i+1} : A, P_{i+2} : B, \dots$  is excluded by  $R_6$ ).

**i)** Suppose  $A \gg^{\text{Pref}} B$ .

In case 1),  $l'$  terminates in  $P_i : A$ , and so the line status of  $P_i : A$  is *in*. Hence the line status of every move by  $P$  in  $l'$  is *in*, every move by  $O$  is *out*.

In case 2) the line-status of  $O_{i+2} : A$  is *in* since by  $R_{RP}$  proponent does not reply to  $O_{i+2}$ . Hence, the line-status of moves  $P_{i+1} : B$  and  $O_i : A$  are *out* and *in* respectively. The line-status of  $O_i : A$  in  $l'$  ending in  $O_i : A$  remains *in*. Hence, the line status of moves in  $l'$  are the same as in  $l$ .

In case 3) the line-status of  $O_{i+2} : B$  is *in* given  $R_{RP}$ . Hence,  $P_{i+1} : A$  is *out*,  $O_i : B$  is *in*. The line status of  $P_{i+1} : A$  in  $l'$  ending in  $P_{i+1} : A$ , and so every move by  $P$  in  $l'$ , now changes to *in*, and  $O_i : B$ , and so every move by  $O$  in  $l'$ , changes to *out*.

**ii)** Suppose it is not the case that  $B \gg^{\text{Pref}} A$ .

In case 2),  $l$  is pruned to  $l'$  ending in  $O_i : A$ , and the line status of  $O_i : A$  in  $l$  and  $l'$  is *in*. Hence, the line status of moves in  $l'$  are the same as in  $l$ .

Given **i**) - **ii**), then by lemma 1 the dialogical status of  $m_1$  is *in* in  $\text{resolve}(d, \text{Pref})$

The above result is of practical as well as theoretical interest. Preferences only need to be applied to decide the issue when the current winner is the opponent. Furthermore, the restriction on application of preferences to symmetric attacks is satisfied by a number of formalisms. For example, logic programming formalisms such as [9] and [5], that

underpin the CARREL system [10]. In these formalisms,  $A$  asymmetrically attacks  $B$ , only if  $A$  claims (proves) what was assumed non provable (through negation as failure) by a premise in  $B$ .  $A$  then defeats  $B$  irrespective of whether  $B$  is preferred to  $A$ .

**Example 3** For the dialogue in figure 1a), suppose that in the underlying logic any two arguments with logically contradictory conclusions symmetrically attack. Suppose  $O$ 's argument  $C$  for  $\neg q$  where  $\neg$  in  $\neg m$  denotes negation as failure (' $m$  is not provable'). Now suppose:

- $P$  replies to each instance of  $C$  with the move *argue*  $m$  since  $f$ , where  $m$  since  $f$  asymmetrically attacks  $C$  in the underlying logic.

- $P$  replies to  $\neg p$  since  $u$  with *why*  $u$ .

$p$  since  $q, r$  is now *in* and remains *in* irrespective of the preferences between the symmetrically attacking pairs of arguments for  $p$  and  $\neg p$ , and  $q$  and  $\neg q$

### 3.4. Non-repetition with symmetric attacks

Finally, a further result can be proven if all attacks between arguments are symmetric, whether dialogues are rational or not. Consider in addition to  $R_1 - R_8$ , rule  $R_{10}$  defining liberal ‘non-repetition’ dialogues:

$R_{10}$ : if  $d$  contains a line with  $O_i : \text{argue } A; P_{i+1} : \text{argue } B$  then  $m$  does not reply to  $P_{i+1}$  with *argue*  $A$ .

This says that  $O$  may not move  $A$  in reply to argument  $B$  if  $B$  in turn is a reply to  $A$  (for  $P$  this is already excluded by  $R_6$ ). It can now be shown that after resolution the result will be same with our without this new protocol rule.

**Proposition 3** Let  $d$  be a liberal dialogue where  $(A, B) \in \mathcal{R}$  implies  $(B, A) \in \mathcal{R}$  in the underlying logic. Let  $d'$  be a liberal non-repetition dialogue, obtained from  $d$  by pruning every line at a move that violates  $R_{10}$ . Then after pruning both dialogues have the same winner.

**Proof:** We only need to consider cases where a line  $l$  in  $d$  contains  $m_1, \dots, O_i : A; P_{i+1} : B$ . Irrespective of whether we have  $O_{i+2} : A$  or not, if  $A \gg^{Pref} B$  then both prunings yield lines ending in  $O_i : A$ , if  $B \gg^{Pref} A$  then both prunings yield lines ending in  $P_{i+1} : B$ , and if it is not the case that  $B \gg^{Pref} A$ , then both prunings yield lines ending in  $O_i : A$ .

Note that the above result does not hold if we generalise  $R_{10}$  to preclude repetition of argue moves by  $O$  in the same line (i.e., as  $R_6$  does for  $P$ ). Finally, note that non-repetition protocols not only ensure shorter dialogues, but are also more ‘realistic’ in the sense that it is somewhat counter-intuitive for a player to repeat an argue move that it has already submitted.

## 4. Conclusions

In this paper we have described a procedure for applying preferences to dialogues in [7]’s general framework. Preference information unavailable at the time of a dialogue can then be used to determine the winner of the dialogue after termination. We described protocols that account for arguments attacking rather than defeating their targets, and where the subsequent resolution of a dialogue graph based on preference information yields a

result that would have been obtained by the defined defeat relation. We also showed that if preferences are applied only to symmetric attacks, then if proponent is currently winning the dialogue, then he wins irrespective of the preference relation applied. Requirements for applying preferences to dialogue graphs were highlighted by implementations of protocols instantiating the framework that have been deployed in the CARREL system [10], and is also relevant for models of legal adjudication procedures [8]. Future work aims to extend the former implementations to enable application of preferences in the manner described in this paper.

The framework of [7] assumes that backward extension does not weaken arguments, thus presupposing the ‘last link’ evaluation of argument strength [9]. This might suggest that preferences cannot be applied to dialogues formalised in the ‘Toulouse-Liverpool’ approach [2,6], where the ‘weakest link’ valuation of arguments violates this principle. However, this approach does not effectively allow for backward extending of arguments. Participants make claims, and when queried are required to defend these claims with Dung acceptable arguments constructed from their shared commitments and individual belief bases. Thus any such argument moved will already be backward extended to the extent that it can be. Any premise challenged will elicit an alternative argument for that premise. However, application of preferences to such dialogues will require extending the pruning mechanisms described in this paper. This is because the dialogues in [2,6] do not explicitly model argue moves replying to argue moves; rather, this reply relation is implicit, in that claims or assertions of propositions reply to each other, where the arguments for these assertions are elicited by why moves.

**Acknowledgements** This work was partly funded by the EU 6th framework projects ASPIC (FP6-002307) and CONTRACT (FP6-034418).

## References

- [1] L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34(1-3):197–215, 2002.
- [2] L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proc. 4th International Conference on MultiAgent Systems (ICMAS-00)*, 31–38, Boston, MA, 2000.
- [3] ASPIC. Deliverable d2.1: Theoretical frameworks for argumentation. [http://www.argumentation.org/Public\\_Deliverables.htm](http://www.argumentation.org/Public_Deliverables.htm), June 2004.
- [4] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and  $n$ -person games. *Artificial Intelligence*, 77:321–357, 1995.
- [5] S. Modgil, P. Tolchinsky, and U. Cortés. Towards formalising agent argumentation over the viability of human organs for transplantation. In *4th Mexican Int. Conf. on Artificial Intelligence*, 928–938, 2005.
- [6] S. Parsons, M. Wooldridge, and L. Amgoud. An analysis of formal interagent dialogues. In *Proceedings of the First International Conference on Autonomous Agents and Multiagent Systems (AAMAS-02)*, 394–401, 2002.
- [7] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15:1009–1040, 2005.
- [8] H. Prakken. A formal model of adjudication. In S. Rahman, editor, *To appear in: Argumentation, Logic and Law*. Springer Verlag, Dordrecht, 2007.
- [9] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7:25–75, 1997.
- [10] P. Tolchinsky, U. Cortés, S. Modgil, F. Caballero, and A. Lopez-Navidad. Increasing Human-Organ Transplant Availability: Argumentation-Based Agent Deliberation. *IEEE Special Issue on Intelligent Agents in Healthcare*, 21(6):30–47, 2006.
- [11] P. Tolchinsky, S. Modgil, U. Cortés, and M. Sánchez-Marré. CBR and argument schemes for collaborative decision making. In *Proc. 1st International Conference on Computational Models of Argument*, 71–82, 2006.

# A Methodology for Action-Selection using Value-Based Argumentation

Fahd Saud Nawwab<sup>1</sup>, Trevor Bench-Capon, Paul E. Dunne

*Department of Computer Science, University of Liverpool, Liverpool, UK*

**Abstract** This paper describes a method for decision making using argumentation. The method is intended to produce the decision considered most likely to promote the agent's aims and aspirations. First, the problem situation is formulated in terms of an Action-Based Alternating Transition System, representing the actions available to the agents relevant to the situation, and their consequences, taking into account the possible effects of the choices of the other relevant agents. Next, arguments are constructed by instantiating an argumentation scheme designed to justify actions in terms of the values they promote and subjecting these instantiations to a series of critical questions to identify possible counter arguments. The resulting arguments are then organized into a Value-Based Argumentation Framework (VAF), so that a set of arguments acceptable to the agent can be identified. Finally the agent must select one of the acceptable actions to execute. The methodology is illustrated through the use of a detailed case study.

**Keywords.** Decision making based on argumentation, Reasoning about action using argument, Applications.

## Introduction

When choosing what to do in a given situation an agent needs not only to identify its options and their likely effects, but to take into account factors outside of its control, such as the choices of other agents which can change the effects of its own actions. Moreover the choice will be determined by the short term and long term aims and aspirations of the agent, and perhaps also by its emotions and temperament [6]. These factors will differ from agent to agent, and so different agents may rationally decide to pursue different courses of action. This paper will present a methodology for decision making for use by an autonomous agent.

In selecting an action an agent needs to have regard to a range of considerations. It must take a view of the relevant features of the current situation in so far as they affect the range of options available, and the effects of the actions selected. The reasons why the effects are desirable and the priority given to the various desires by the agent will be important. The impact of any uncertainties in the current situation on the available actions and their effects needs to be considered. Performing an action will often mean that some other action cannot be performed, while other actions may be enabled. The side effects of actions need to be considered: in some cases any gains may be outweighed by losses. Conversely there may be beneficial side effects, increasing the

---

<sup>1</sup> Corresponding Author: E-Mail: fahad@csc.liv.ac.uk

attractiveness of the choice. Any method for action selection needs to take these considerations, and more, into account.

Our analysis is based on the use of argumentation schemes as a presumptive justification of action: in particular we use the argument scheme for justifying actions described in [3]. This scheme is also designed to allow for considerations stated in the preceding paragraph to be reasoned about through posing critical questions challenging the justification which will be seen as counter arguments. Resolving the conflicts between these arguments and their counter arguments will ensure that the considerations are given their due weight. Resolution is achieved using a Value-Based Argumentation Framework [4], which relates acceptance of the justifying arguments to the particular priority order the agent gives to the values motivating the action and so allows for the representation of subjective choice.

The approach to the problem can be seen as involving five stages:

- *Formulating the Problem:* produce a formal description of the problem scenario to give all possible actions, values and all related factors that may influence the decision. This will be accomplished through an Action Based Alternating Transition System (AATS) as in [2].
- *Determining the Arguments;* on the basis of the AATS, arguments providing justifications of the various available actions are provided by instantiating the argument scheme introduced in [3]. Counterarguments are identified using a subset of the critical questions of [3], as interpreted in terms of an AATS in [2].
- *Building the Argumentation Framework:* In this step the arguments and attacks between them identified in the previous step are organized into an Argumentation Framework. Because the argument scheme used associates arguments with the values they promote or demote, arguments can be annotated with these values, yielding a Value Based Argumentation Framework (VAF) [4].
- *Evaluating the Argumentation Framework.* As described in [4], the arguments within VAFs are determined as acceptable or not with respect to a specific audience, characterized by the ordering on values subscribed to by the agent making the choice.
- *Sequencing the Actions.* The set of actions deemed to be acceptable to the agent in the previous stage must now be put into a suitable order in which they should be performed.

The first section of this paper has given a brief background summarizing the approach being used. The second section will define the various elements that make up the proposed methodology. The third section will introduce the case study and provide a detailed working through each of the five steps. Finally, we present some concluding remarks, observations and possible enhancements to this work.

## 1. Decision Making Framework

In this section we will describe the techniques we use in each of the five steps.

### 1.1 Formulating the Problem

In step one; the problem is formulated as an Action-Based Alternating Transition System (AATS). AATS were introduced in [10] as a foundation to formally describe a system in which several agents combine to determine the transition between states. In [10] an AATS with  $n$  agents is an  $(n+7)$  tuple; This was then extended by [2] to include the notion of values so that each agent has a set,  $Av$ , of values drawn from an underlying set of values  $V$ , and every transition from the set  $Q$  of different possible states may either promote, demote, or be neutral with respect to those values.

#### Definition 1: AATS:

A joint action  $j_C$  for set of agents, termed a *coalition*,  $C$ , is a tuple  $\langle \alpha_1, \dots, \alpha_k \rangle$ , where for each  $\alpha_j$  (where  $j \leq k$ ) there is some  $i \in C$  such that  $\alpha_j \in Ac_i$ . Moreover, there are no two different actions  $\alpha_j$  and  $\alpha_{j'}$  in  $j_C$  that belong to the same  $Ac_i$ . The set of all joint actions for coalition  $C$  is denoted by  $J_C$ , so  $J_C = \prod_{i \in C} Ac_i$ . Given an element  $j$  of  $J_C$  and an agent  $i \in C$ ,  $i$ 's action in  $j$  is denoted by  $j_i$ .

As extended by [2], an AATS is a  $(2n+8)$  tuple  $S = \langle Q, q_0, Ag, Ac_1, \dots, Ac_n, Av_1, \dots, Av_n, p, \tau, \Phi, \pi, \delta \rangle$  where:

$Q$  is a finite, non-empty set of *states*

$q_0 = q_x \in Q$  is the *initial state*

$Ag = \{1, \dots, n\}$  is a finite, non-empty set of *agents*

$Ac_i$  is a finite, non-empty set of actions, for each  $i \in Ag$  where  $Ac_i \cap Ac_j = \emptyset$  for all  $i \neq j \in Ag$ ;

$Av_i$  is a finite, non-empty set of values  $Av_i \subseteq V$ , for each  $i \in Ag$ .

$p : Ac_{Ag} \rightarrow 2^Q$  is an *action precondition function*, which for each action  $\alpha \in Ac_{Ag}$  defines the set of states  $p(\alpha)$  from which  $\alpha$  may be executed;

$\tau : Q \times J_{Ag} \rightarrow Q$  is a partial *system transition function*, which defines the state  $\tau_{(q,j)}$  that would result by the performance of  $j$  from state  $q$  - note that, as this function is partial, not all joint actions are possible in all states (cf. the precondition function above);

$\Phi$  is a finite, non-empty set of *atomic propositions*;

$\pi : Q \rightarrow 2^\Phi$  is an interpretation function, which gives the set of primitive propositions satisfied in each state: if  $p \in \pi(q)$ , then this means that the propositional variable  $p$  is satisfied (equivalently, true) in state  $q$ .

$\delta : Q \times Q \times Av_{Ag} \rightarrow \{+, -, =\}$  is a *valuation function* which defines the status

(promoted (+), demoted (-) or neutral (=)) of a value  $v_u \in Av_{Ag}$  ascribed by the agent to the transition between two states:  $\delta(q_x, q_y, v_u)$  labels the transition between  $qx$  and  $qy$  with one of  $\{+, -, =\}$  with respect to the value  $v_u \in Av_{Ag}$ .

To represent a particular problem we first identify a set of propositions which we consider relevant to the scenario. Each model of this set of propositions will be a potential state of the system. Next, we identify the relevant agents and the different possible actions the agents can perform, and how these will move between states, each

transition representing a joint action of the agents involved. Finally we identify the values and relate them to the transitions between states.

## 1.2 Determining the Arguments

Our method of justifying actions is in terms of presumptive justification through the instantiation of an argument scheme, followed by a process of critical questioning to see whether the presumption can be maintained, as described in [9]. Specifically we use the argument scheme presented in [3] which extends the sufficient condition scheme of [9] to enable discrimination between the effects of an action (the *consequences*), the desired effects (the *goal*) and the reason why these effects are desired (the *value*).

Thus our argument scheme is: *in the current state the agent should perform action A to reach a new state in which goal G is true, promoting value V.*

In [2] a realization of this scheme in terms of an AATS is given:

### **Definition 2:**

The initial state  $q_0 = q_x \in Q$ , Agent  $i \in Ag$  should perform  $\alpha_i$ , consistent with joint action  $j_n \in J_{Ag}$  where  $j_{ni} = \alpha_i$ , such that  $\tau(q_x, j_n) = q_y$ , where  $p_a \in \pi(q_y)$  and  $p_a \notin \pi(q_x)$ , such that for some  $v_u \in A_{vi}$ ,  $\delta(q_x, q_y, v_u)$  is +.

Critical questions are now used to address the factors which may lead to the presumptive justification being overturned. In [3] sixteen critical questions were identified, but in any given scenario not all of them will be relevant. For our purposes we need to consider only six here as some of the questions concerns problems of language or epistemic matters which do not apply to a single agent. These are defined in terms of an AATS in Definition 3.

### **Definition 3:** Relevant Critical Questions.

CQ1: Are the believed circumstances true?

$q_0 \neq q_x$  and  $q_0 \notin \rho(\alpha_i)$ .

CQ11: Does doing the action preclude some other action which would promote some other value?

In the initial state  $q_x \in Q$ , if agent  $i \in Ag$  participates in joint action  $j_n \in J_{Ag}$ , then  $\tau(q_x, j_n) = q_y$  and  $\delta(q_x, q_y, v_u)$  is +. There is some other joint action  $j_m \in J_{Ag}$ , where  $j_n \neq j_m$ , such that  $\tau(q_x, j_m) = q_z$ , such that  $\delta(q_x, q_z, v_w)$  is +, where  $v_u \neq v_w$ .

CQ2: Assuming the circumstances, does the action have the stated consequences?

$\tau(q_x, j_n)$  is not  $q_y$ .

CQ7: Are there alternative ways of promoting the same value?

Agent  $i \in Ag$  can participate in joint action  $j_m \in J_{Ag}$ , where  $j_n \neq j_m$ , such that  $\tau(q_x, j_m) = q_z$ , such that  $\delta(q_x, q_z, v_u)$  is +.

CQ8: Does doing the action have a side effect which demotes the value?

In the initial state  $q_x \in Q$ , if agent  $i \in Ag$  participates in joint action  $j_n \in J_{Ag}$ , then  $\tau(q_x, j_n) = q_y$ , such that  $p_b \in \pi(q_y)$ , where  $p_a \neq p_b$ , such that  $\delta(q_x, q_y, v_u)$  is -.

CQ9: Does doing the action have a side effect which demotes some other value?

In the initial state  $q_x \in Q$ , if agent  $i \in Ag$  participates, in joint action  $j_n \in J_{Ag}$ , then  $\tau(q_x, j_n) = q_y$ , such that,  $\delta(q_x, q_y, v_w)$  is -, where  $v_u \neq v_w$ .

### 1.3 Building the Relationship Model

The previous step yields a number of arguments, associated with values, and a set of attack relations between them. These can be organized into a Value Based Argumentation Framework (VAF). A VAF is defined in [5] as:

**Definition 4:** Value based Argumentation Framework.

A triple  $\langle H(X, A), V, \eta \rangle$ , where  $H(X, A)$  is an argumentation framework,  $V = \{v_1, v_2, \dots, v_k\}$  a set of  $k$  values, and  $\eta : X \rightarrow V$  a mapping that associates a value  $\eta(x) \in V$  with each argument  $x \in X$ . A specific audience,  $\alpha$ , for a VAF  $\langle H, V, \eta \rangle$ , is a total ordering of the values  $V$ . We say that  $v_i$  is preferred to  $v_j$  in the audience  $\alpha$ , denoted  $v_i >_\alpha v_j$ , if  $v_i$  is ranked higher than  $v_j$  in the total ordering defined by  $\alpha$ .

**Definition 5:**

Let  $\langle H(X, A), V, \eta \rangle$  be a VAF and  $\alpha$  an audience.

- a. For arguments  $x, y$  in  $X$ ,  $x$  is a successful attack on  $y$  (or  $x$  defeats  $y$ ) with respect to the audience  $\alpha$  if:  $\langle x, y \rangle \in A$  and it is not the case that  $\eta(y) >_\alpha \eta(x)$ .
- b. An argument  $x$  is acceptable to the subset  $S$  with respect to an audience  $\alpha$  if: for every  $y \in X$  that successfully attacks  $x$  with respect to  $\alpha$ , there is some  $z \in S$  that successfully attacks  $y$  with respect to  $\alpha$ .
- c. A subset  $R$  of  $X$  is conflict-free with respect to the audience  $\alpha$  if: for each  $\langle x, y \rangle \in R \times R$ , either  $\langle x, y \rangle \notin A$  or  $\eta(y) >_\alpha \eta(x)$ .
- d. A subset  $R$  of  $X$  is admissible with respect to the audience  $\alpha$  if:  $R$  is conflict free with respect to  $\alpha$  and every  $x \in R$  is acceptable to  $R$  with respect to  $\alpha$ .
- e. A subset  $R$  is a preferred extension for the audience  $\alpha$  if it is a maximal admissible set with respect to  $\alpha$ .

### 1.4 Evaluating the Model

Having constructed a VAF, the next step is to evaluate the attacks and determine which arguments will be acceptable to the agent. The strength of each argument is determined by the values associated with it. Given the ordering on values desired by the agent, we can determine what arguments will be acceptable to the agent, and determine the preferred extension with respect to the audience endorsed by the agent. This preferred extension, which will be unique and non-empty, represents the maximal set of acceptable arguments for that agent.

### 1.5 Sequencing the Actions

The four previous steps have identified a set of actions acceptable to the agent given its priorities with respect to values. These actions are acceptable in the sense that they have survived the critique provided by the posing of critical questions and have no attackers preferred to them. Often this set will contain several actions, any of which could be beneficial to perform in the current state. We suggest that these should be sequenced in terms of *safety*, by which we mean that unexpected consequences will not prevent the other actions in the set being performed; *opportunity*, by which we mean that the performance of an action may enable some desirable action which is not

available in the current state to be executed subsequently; and *threat*, where a potentially bad side effect is brought into play.

## 2. Example Application

### 2.1 Formulating the Problem

Our case study concerns a problem which faces university Heads of Department, and reflects the need to balance costs, Departmental and individual interests. Our agent is a Head of Department (HoD) in a university, and has requests relating to travel funding to attend two specific conferences. He has three potential candidates and needs to decide which of them to send. Students 1 (S1) and 2 (S2) are new students. S1 is asking to go to a nearby conference, which will be cheaper financially, S2 is asking for a different conference which will cost more, but S2 has prepared a good paper that might help the department's reputation. Student 3 (S3) is an older, experienced, student asking to be sent to the local conference and, although she has not prepared a paper, she is an excellent networker who is likely to impress other delegates and so present the department in a good light. The conferences are on different topics, so S2's paper would not be suitable for the local conference, but both conferences are of equal standing. The budget will only allow two students to be sent.

Now, in the rest of this section we will set the different properties to allow the representation of the problem of our example as an AATS.

#### 2.1.1 Propositions and Actions

We will first consider the different propositions that the agent will take into account in his decision making: whether there are currently funds available in the budget (Budget); whether a student can be sent to attend (Attendance S(1-3)), whether the student has written a paper (Paper S(1-3)) and, finally, whether the student has attended a conference before (Previous S(1-3)).

Now, we define all possible actions that the HoD can take in all circumstances. Those are either to ask any one of the three students to write a paper Write(S1), Write(S2), Write(S3), or to agree to send a student to the requested conference Send(S1), Send(S2), Send(S3). These actions may change the state of Paper(Si) or Attendance(Si) respectively.

#### 2.1.2 Values

Now, we list the values significant for the HoD. We, then, link those values to the various transitions. Table 1 shows the different values that will be promoted or demoted in accordance with the changes in propositions.

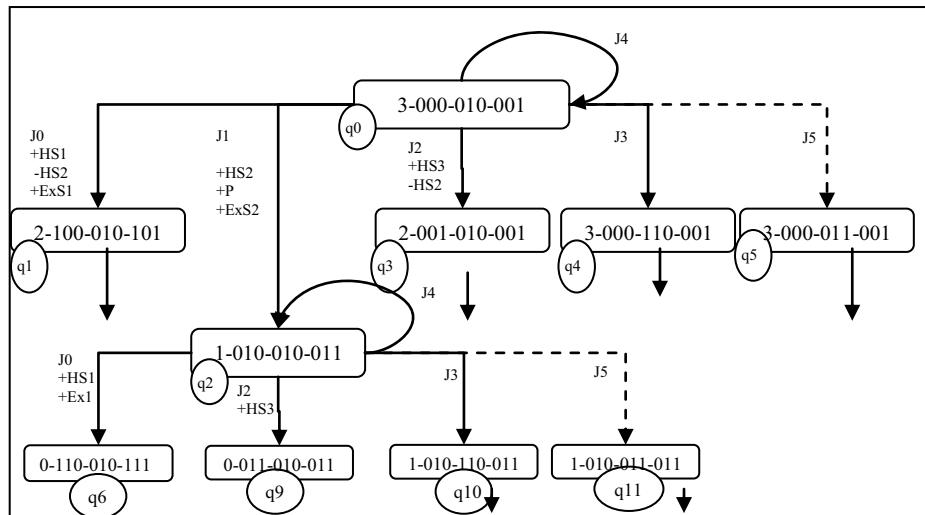
Value	Short	Promoted/Demoted if:
Happiness	H(Si)	Promoted if Si attends
Happiness	H(Si)	Demoted if Si has written a paper and does not attend
Publication	P	Promoted if Si attends having written a paper
Experience	E(Si)	Promoted if Si has not attended before, and attends
Esteem	Est	Promoted if Si has attended a previous conference, has a paper and attends

Table 1: Values relevant to the audience (HoD)

By esteem, we mean the general enhancement of the reputation of the Department that comes from an impressive individual making an impact at a conference and raising the profile of the Department's research, the research links established, and such like. Note that happiness and experience are relative to individual students, whereas the other values are relative to the Department, although realized through properties of the individual students.

### 2.1.3 State Format

The states will be presented as follows: Budget, Attendance, Paper and Previous (B-XXX-XXX-XXX) where each X will be either 1 or 0 depending on whether or not the corresponding proposition is true in that state. Before we move into modelling the state transitions, let us look at the initial state  $q_0$ . Budget is set to 3: the cheaper conference costs 1 and the expensive conference 2, so that we can send at most two students.  $S_1$  and  $S_3$  will consume 1 point from the budget whenever chosen whereas  $S_2$  will consume 2 points.  $S_3$  has already attended a previous conference and  $S_2$  has a paper ready. Thus,  $q_0 = (3-000-010-001)$ . Figure 1 shows the initial state and some example transitions from that state. Action  $J_0$  is  $\text{Send}(S_1)$ ,  $J_1$  is  $\text{Send}(S_2)$ ,  $J_2$  is  $\text{Send}(S_3)$ . Where a paper is requested and written, we have  $J_3$  for  $S_1$  and  $J_5$  for  $S_3$ , while  $J_4$  represents a request which does not result in a paper. The transitions are also labeled with the values they promote or demote. Budget = 0 represents a terminal state, since no further actions are possible.



**Figure 1: Transitions from initial state: States with exit arrows have successor states.**

### 2.1.4 Uncertainties

Consequences of actions are not always entirely predictable and actions are executed hoping for a certain result which often will not come about because it has

dependencies on other actions performed by other agents. When an agent performs an action where the results solely depend on itself, as with sending a student to a conference, it is very easy to assume the resulting state. When, however, the HoD asks a student to write a paper, the student may or may not succeed. And so he cannot be certain which state will be reached, this is shown in Figure 1 with a dashed line. There may also be uncertainty about the initial state: in Figure 1 we assume that S2 is the only student who has written a paper. But it may be that the HoD is not sure of this, and any of the three students might have actually written a paper. This gives us eight different possible states, any of which could be the initial state. If an action is performed in a state other than the one assumed, the state reached may be different. These uncertainties will be considered through the mechanism of critical questions.

We now move to the second step where we start building arguments for and against performing the various actions.

## 2.2 Determining the Arguments

The AATS will allow us to evaluate each action at every state and relate the actions to propositions and the values they promote. Table 2 shows the arguments that can be made for performing an available action in the initial state.

Arg	In State	Action	To get to State	Realize Goal				Promoting			
				Budget	Attend	Paper	Prev	H	P	E	Est
Arg1	Q0	J0	Q1		S1		S1	S1			
Arg2	Q0	J0	Q1		S1		S1			S1	
Arg3	Q0	J1	Q2		S2		S2	S2			
Arg4	Q0	J1	Q2		S2		S2		S2		
Arg5	Q0	J1	Q2		S2		S2			S2	
Arg6	Q0	J2	Q3		S3			S3			
Arg7	Q0	J3	Q4			S1					
Arg8	Q0	J5	Q5			S3					

**Table 2: Arguments from state q0**

We can see from the table how these arguments differ with respect to the values promoted. The next step now is to use critical questions to identify which arguments are open to attack. We will use the identifying labels of critical questions as in [2].

## 2.3 Building the Relationship Model

### 2.3.1 CQ1: Are the stated circumstances true?

This question arises in this example from the fact that although the HoD believes that the initial state is 3-000-010-001 (q0) where S2 has written a paper, he cannot actually be sure that the other students have not also written papers and cannot be absolutely certain that S2 has in fact written a paper. This results in there being eight different possible initial states. So, in this case, all the arguments in Table 2 are open to this attack.

The agent could assume that all states are possible and build up the argumentation model with all the possible states in mind. This will result in multiple Preferred Extensions (PEs), one for each possible initial state. The common elements in all PEs will then represent justifications of actions which are unaffected by the uncertainties

with respect to what is true in the initial state. Should there be no arguments common to all the PEs, it would be necessary to make choices as to what is to be believed in the original situation: for example it may be considered very unlikely that S2 would have misinformed the HoD about the status of his paper. It is also possible that the agent is able to confirm his beliefs before proceeding to make decisions. In our example this is the case: the HoD can ask S2 to show him the paper, and ask the others if they have a paper ready. In what follows, therefore, we can assume that the HoD is able to confirm his beliefs, and so objections arising from CQ1 can be discounted. In general, however, where complete information is not attainable, the question is an important one.

### *2.3.2 CQ11: Does the action preclude some other action which would promote some other value?*

S1 or S3 are sent to the conference without being asked to write a paper, the chance to promote publication is lost. Moreover, this will also lose the chance to promote esteem, which requires S3 to be sent with a paper written. Thus Arg1, Arg2, and Arg6 are all attacked by an argument, A1a, in that they prevent the promotion of publication. Arg6 is also attacked by an argument, Arg6a, that it precludes the promotion of esteem.

### *2.3.3 CQ2: Does the Action have the stated consequences?*

This question occurs when we need to consider joint actions: cases where the agent is not in sole control of the state reached. In our example, this is represented by the possibility of the request to write a paper not being met. Thus Arg7 and Arg8 are attacked by Arg7a, that the joint action might turn out to be J4.

### *2.3.4 CQ8: Does the action have side effects which demote the value?*

Sending any of the students other than S2, will demote the happiness of S2, since he has already written a paper. Supposing the HoD is impartial and so indifferent as to which student happiness is promoted in respect of, this will give an argument, Arg1b, against Arg1 and Arg6: while these arguments promote happiness, the actions they justify also demote it.

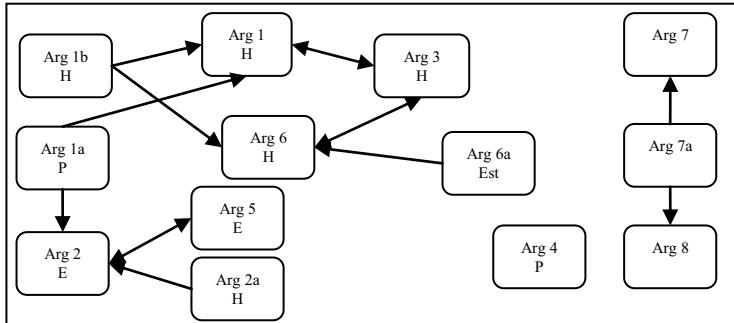
### *2.3.5 CQ9: Does the action have side effects which demote some other value?*

If a student who has written a paper is not sent, the happiness of that student will be demoted. This provides the basis for an argument against doing any action which involves not sending S2 for any reason other than the promotion of happiness. Thus Arg2 is subject to an attack from Arg2a since it would demote S2's happiness.

### *2.3.6 CQ7: Are there other ways to promote the same value?*

Both Arg2 and Arg5 are based on the promotion of experience. This question indicates that they attack one another. Similarly happiness can be promoted by any of Arg1, Arg3 and Arg6. These also mutually attack, therefore.

Now, we have identified all possible attacks from the different arguments we are able to form the value based argumentation framework. Figure 2 is a graphical representation of the framework in the example.



**Figure 2: Value Based Argument Framework for q0**

#### 2.4 Evaluating the Model

From the VAF in Figure 2 we can see that Arg4 has no arguments attacking it. Thus it will appear in every preferred extension, irrespective of the ranking of values. The status of Arg7 and Arg8 depend on whether the HoD is confident that the papers will be written if requested. Suppose his confidence is sufficient, and so Arg7 and Arg8 are acceptable. In order to determine which of the remaining arguments are acceptable, the values that the HoD wishes to promote at the particular time needs to be ordered. Suppose that the value ordering is as follows: Esteem > Publication > Experience > Happiness. This gives us the ability to resolve the conflicts that we have in the model by eliminating unsuccessful attacks. Arg1b will defeat Arg1 and Arg6, leaving Arg3 for the preferred extension. Although Arg2 is not defeated by Arg2a, it is defeated by Arg1a, and so Arg5 survives. Thus our preferred extension is {Arg1a, Arg1b, Arg2a, Arg3, Arg4, Arg5, Arg6a, Arg7, Arg8}. In terms of actions, sending S2 can be justified, as can be requesting a paper from S1 and S3. The arguments in the preferred extension which do not justify actions are there to justify the rejection of other arguments.

#### 2.5 Sequencing the Actions

The last step identified three actions, which would move to q2, q4, or q5, although if the confidence in the student's ability to produce a paper was misplaced, the state would remain q0. Now some sensible sequence for these actions must be chosen. This choice will need to consider both uncertainty about outcomes (Section 2.1.4) and eventually reaching the state best for the agent, and will require looking ahead to consider what is possible in the states that would result from our action.

There are three issues we should consider here. First we need to consider whether the action is *safe*, in the sense that if it fails we do not move to a state where our other desirable actions are no longer possible. In our example, all our actions are safe, since those that can fail simply return us to the initial state. Next we must consider *opportunities*: what additional values can be promoted in the next state? If we ask S1 to write a paper, we have the possibility of promoting publication (the chance to promote S1's experience already exists as Arg2), and, although we can already promote this by sending S2, S1's publication will be an additional benefit. But if we ask S3 to write we

can create the chance to promote esteem, as well as the additional publication. But there are also *threats*: if S1 and S3 write papers and are not sent, they will be unhappy. Since we have said we prefer esteem to experience, we will prefer the opportunities created by requesting a paper from S3, and so will prioritise this action over asking S1. Should we do this before sending S2? In the particular example this seems to not matter. We might, however, decide that if we sent S2, we would demotivate S2 and so reduce the likelihood of her producing a paper. Such calculation of the probability of the success of actions is outside the scope of this paper, but would be a suitable topic for further investigation. Suppose, however, we make this judgement: then we should request a paper from S3 before sending S2. If S3 does write the paper, we will move to another state in which the recommended actions will be to send S2 and S3. If, however, S3 does not produce a paper: now we have no possibility of promoting S3, and no threat of making S3 unhappy, and so we should request a paper from S1, in the hope of making an opportunity to promote publication. If S1 does write, we should then send S1 and S2: and even if S1 does not write a paper he should be sent as this will still promote experience in respect of S1.

Note that had S3 been interested in the expensive conference, the situation represented in Figure 2 would differ, since Arg6a would now, via CQ11, attack, successfully given our value order, Arg3, Arg4 and Arg6, since we can send at most one student to the expensive conference. This would make requesting a paper from S3 the only choice. Note also that had the HoD preferred experience to esteem, he would ask S1 rather than S3 to write a paper, and send S1 whether or not a paper is written: once it has been decided to send S2, there is a straight choice between S1 and S3 so that sending S3 will preclude the promotion of experience, and so Arg6 will have an attacker based on CQ11.

### 3 Conclusion

In this paper we have used a case study to present a methodology for decision making in a situation where a number of competing interests need to be balanced. The example provides an illustration of how a set of relevant arguments can be generated and evaluated in accordance with the particular preferences of the decision maker. The example has drawn attention to two matters in particular: the need to decide upon the most appropriate sequence for actions which are justifiable in a given situation, and the need to consider uncertainty with respect to actions which depend for their effect on what other agents will choose to do. With regard to sequencing actions we can see that even if the execution of a sequence of actions would result in the same final state there may still be reasons for choosing one ordering rather than another. This is because the paths taken to that state will differ, and so the intermediate states may give rise to different opportunities and threats, and because should the effects of actions be not what was expected, there will be differences in the ability to recover from these setbacks. We also need to take account of uncertainty: our confidence in what the other agent will do may differ from state to state. Thus different sequences may increase or diminish our confidence that the joint action will be as desired, and we should sequence our actions so as to achieve as much confidence as possible when we choose to perform the action.

For future work, one issue worth exploring would be a technique for gauging uncertainty of the effects of actions, or working with the uncertainty. There are other interesting directions also. We have assumed that the agent is able to provide a total order on its values, but, as noted in [8], such an ordering often emerges as part of the reasoning process. Possibilities for this would be either to stipulate that some arguments must be made acceptable, as in [5], or to allow reasoning about what the value order should be, as in [7]. Another direction is to envisage the agent as being an automated decision maker acting on behalf of a human decision maker. In such cases, it may be that the agent will make decisions which the human does not endorse: such feedback would suggest that the value order used by the agent should be modified. But the nature of these modifications, and the timing of these modifications is not a straightforward matter: for example it could be that having preferred H(S1) to H(S2) on one occasion, a HoD would feel obliged to use the opposite preference on the next occasion. Seeing the problem in the context of an ongoing series of decisions would require additional considerations to be taken into account. Finally we need to accommodate the fact that these kinds of decisions are not always entirely based on a rational assessment of their pros and cons: emotions can also play a role:

“Emotions and feelings can cause havoc in the process of reasoning under certain circumstances. Traditional wisdom has told us that they can, and recent investigation of the normal reasoning process also reveal the potentially harmful influence of emotional biases. It is thus even more surprising and novel that the absence of emotion and feeling is no less damaging, no less capable of compromising the rationality that makes us distinctively human and allows us to decide in consonance with a sense of personal future, social convention, and moral principle.” [6]

These considerations open further avenues for exploration which would be immensely interesting to explore.

## References

- [1] K. Atkinson. *What should we do? Computational representation of persuasive argument in practical reasoning*. PhD thesis, University of Liverpool, 2005.
- [2] K. Atkinson and T. Bench-Capon. Action based alternating transition systems for arguments about action. In *Proceedings of the Twenty Second Conference on Artificial Intelligence* (AAAI 2007), Vancouver, Canada, pp. 24-29. AAAI Press, Menlo Park, CA, USA.
- [3] K. Atkinson, P. McBurney, and T. Bench-Capon. Computational representation of practical arguments. *Synthese*, 152(2):157-206, pages 191–240, 2006.
- [4] T. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
- [5] T.J.M. Bench-Capon, S. Doutre and P.E. Dunne. Audiences in Argumentation Frameworks. *Artificial Intelligence*, Vol 171 no 1, pp42-71, 2007.
- [6] A. Damasio. *Descartes' Error*. G P Putnam and Sons, 1994.
- [7] S. Modgil. *Value Based Argumentation in Hierarchical Argumentation Frameworks*. In: Proc. 1st International Conference on Computational Models of Argument. IOS Press, Amsterdam, pp 297-310.
- [8] J. R. Searle. *Rationality in Action*. The MIT Press, Cambridge, Massachusetts, 2003.
- [9] D. N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1996.
- [10] M. Wooldridge and W. van der Hoek. On obligations and normative ability: Towards a logical analysis of the social contract. *J. Applied Logic*, 3(3-4):396–420, 2005.

# Semantics for Evidence-Based Argumentation

Nir Oren <sup>a,\*</sup>, Timothy J. Norman <sup>b</sup>

<sup>a</sup> King's College London, London, UK

<sup>b</sup> University of Aberdeen, Aberdeen, Scotland

**Abstract.** The identification of consistent sets of arguments is one of the most important concerns in the development of computational models of argument. Such *extensions* drive the defeasible reasoning process. In this paper we present a novel framework that offers a semantics for argumentation-based reasoning grounded in an intuitive notion of evidence. In building this model of evidence-based argumentation, we first show how evidential support for and attack against one argument by multiple arguments can be represented. Secondly, we propose a new set of extensions, referred to as the evidential preferred, e-stable, and e-grounded extensions. These extensions are similar to those in Dung's original model, but are modified to cater for the notion of evidence. Our approach involves defining a new notion of attack, and showing how acceptability and admissibility follow on from it. These extensions capture the notion that an argument cannot be accepted unless it is supported by evidence, and will therefore increase the utility of argumentation as part of a reasoning system.

**Keywords.** Abstract argumentation frameworks, evidence-based reasoning

## 1. Introduction

Dung's seminal abstract argumentation framework [5] consists of a set of arguments and a binary relation between pairs of arguments. This relation represents the concept of one argument attacking another, and is referred to as the *attacks* relation. For example, consider the following set of arguments: (a) The bridge should be built on solid ground; (b) The bridge should be built at point x, where soft ground exists; (c) The bridge should be built out of wood, not concrete; (d) The bridge should be built out of concrete, not wood; and (e) Building at point x means that concrete should be used, not wood.

These arguments could be represented by the argument framework with arguments  $\{a, b, c, d, e\}$  and attack relation  $\{(a, b), (b, a), (c, d), (d, c), (e, c)\}$  (i.e. the argument that the bridge should be built on solid ground,  $a$ , attacks the argument that bridge should be built at point x,  $b$ , etc.).

Different modes of reasoning are represented by different *extensions*, allowing one to determine what sets arguments a sceptical, or credulous reasoner would deem acceptable. For example, Dung's preferred extension, representing a credulous reasoner, would

---

\*Correspondence to: Nir Oren, Department of Computer Science, King's College London, Strand, WC2R 2LS, United Kingdom. Tel.: +44 (0)20 7848 1631 ; Fax: +44 (0)20 7848 2851 ; E-mail: nir.oren@kcl.ac.uk

consist of the arguments  $\{d, a, e\}$  and  $\{d, b, e\}$ ; these arguments are, in some sense, “consistent” to a credulous reasoner. A sceptical reasoner on the other hand, may only find  $\{d, e\}$  consistent. Deciding what type of extension to use is application dependent. If agents were negotiating, for example, the presence of argument  $d$  in all extensions would mean that they all agree to its relevance. The extensions of an argumentation system are often referred to as the system’s argumentation semantics. The abstract nature of argument in these frameworks allows them to be applied to many domains through instantiation with different underlying logics. While powerful and elegant, Dung’s framework has a number of shortcomings, and various refinements have been proposed.

Looking at the example above, it appears not only as if argument  $e$  attacks argument  $c$ , but that, in some sense, it supports argument  $d$ . This notion of support was recently formalised using bipolar argumentation frameworks (BAFs) [1,2,4].

Another important enhancement of Dung’s original work involves allowing multiple arguments to attack one another [6]. For example, consider the two arguments: (f) Financial considerations override any other considerations; and (g) Financial considerations mean that the bridge should be built at point  $y$ . It is clear that alone, neither argument is able to attack argument  $b$ , but that both arguments, when considered together, combine to create a valid attack against it<sup>1</sup>.

In this paper, we draw inspiration from both BAFs and the work of Nielsen and Parsons, with an eye to providing a rich framework that captures our intuitions of evidential reasoning. Evidential reasoning involves determining which arguments are applicable based on some evidence. Such reasoning arises in many fields of human endeavour, and the formalisation of argument for such domains will, we believe, lead to more powerful reasoning mechanisms. Our first extension involves allowing for the use of multiple arguments not only when attacking an argument, but also in providing evidential support for an argument (in the example above,  $b$  and  $e$  together support argument  $d$ ). Second, we introduce the notion of support into our framework and present a number of new extensions that allow for an intuitive representation of evidential reasoning. Informally, these extensions only accept arguments that are backed up by a chain of evidence. To allow for this, we redefine the concept of attack so that only supported arguments may attack each other. We then show how supported arguments may be directly, or indirectly attacked.

In the next section, we discuss some criticisms that have been levelled at abstract argumentation frameworks that include the notion of support. The following section details our model of evidence-based argumentation, including our notions of acceptability and admissibility, and evidential preferred extensions, evidential stable extensions and evidential grounded extensions. We then use an example in discussing the merits of the model presented in this paper and explore related and future research.

## 2. The Importance of Evidential Support

Introducing the notion of support between arguments within abstract argumentation frameworks has not been met with universal agreement. The criticisms aired can be summarised as follows: (1) representing support at the argument level leaves too many

---

<sup>1</sup>These enhancements are somewhat controversial, as discussed in the next section, with some arguing that they represent different concepts, and should not be used within abstract argument frameworks.

choices regarding how to represent argument; (2) support represents inference, which should be at the level of logic, not argument; and (3) abstract argument frameworks represent support implicitly through the notion of defence; an argument  $a$  supports  $b$  by attacking all attackers of  $b$ . Similar to (3) is claim (4), namely that abstract argument frameworks represent support by default; i.e. the presence of an argument within a framework means that some support for it exists.

The first argument against support could equally be deemed an argument for its inclusion. While a number of representational choices open up, it is possible to get rid of the redundancy that can appear in attack-only argument frameworks. Similarly, by adopting the maxim “represent arguments in the simplest form possible”, one can often eliminate a lot of the choices that may appear. The remaining points are related, as they assume that support is used to somehow infer the case for an argument appearing in either the argument framework, or in its extensions. While this is often the case, particularly in Bipolar Argumentation Frameworks (BAFs), support has another role, namely to allow us to distinguish between *prima facie* and standard arguments. *Prima facie* arguments do not require any sort of support from other arguments to stand, while standard arguments must be linked with at least one *prima facie* argument via a support chain. It is this concept that we formalise in our framework below. Finally, when people reason about argument, they often think of the interactions between support and attack. Therefore, when modelling human argument, it makes sense to have an argument framework that treats the two as equals. BAFs, while allowing for support, use it to model strengthening of conclusions by separate arguments, or occasionally, inference. Arguments may stand with, or without support, meaning that inference in BAFs may take place both internally to the argument, and outside it. The criticisms described above thus apply directly, and we believe legitimately, to Bipolar Argumentation Frameworks. Our notion of support is subtly but importantly different, however; we wish to capture the notion of *evidential support*. We introduce a set of semantics that allow us to reason about arguments that are directly supported by evidence (or are *prima facie* arguments), and arguments that have been inserted into the framework due to a chain of evidential support from a *prima facie* argument. This is a new, and important notion of support.

### 3. Arguments, Attack and Support

As mentioned in the introduction, an argument is accepted only if it is supported through a chain of arguments, each of which is themselves supported. At the head of this chain of supporting arguments is an argument representing support from the environment (written as support from the special argument  $\eta$ ). To represent this notion, we define an evidential argument system as follows:

#### **Definition 1. (Evidential Argumentation Systems)**

An evidential argumentation system is a tuple  $(A, R_a, R_e)$  where  $A$  is a set of arguments,  $R_a$  is a relation of the form  $(2^A \setminus \{\}) \times A$ , and  $R_e$  is a relation of type  $2^A \times A$ , such that within the argumentation system,  $\exists x \in 2^A, y \in A$  such that  $xR_ay$  and  $xR_ey$ . We assume the existance of a “special” argument  $\eta \notin A$ .

The  $R_e$  and  $R_a$  relations encode evidential support and attacks between arguments. An element of the evidential support relation of the form  $\{\eta\}R_ea$  would represent sup-

port by the environment for the argument  $a$ . Within our argument framework, we are interested in seeing which arguments may eventually be considered to hold. Since any argument attacked by the environment will be unconditionally defeated, we believe that it makes no sense to include such arguments, and therefore prohibit the environment from appearing in the attacks relation. Since the environment requires no support,  $\eta$  may not appear as the second element of a member of  $R_e$ .

For an argument  $a$ , if  $\{\eta\}R_e a$ , we say (in a slight abuse of the English language) that  $a$  has environmental support. Environmental support can be used to model defaults (since an argument that has environmental support is true unless attacked), tautologies (if the argument may never be attacked), and arguments that have incontrovertible evidence in their support. We assume that evidence from the environment that cannot be challenged, such as an observation by an infallible sensor, can be thought of as such evidence.

Unlike Bipolar Argumentation Frameworks (BAFs), we allow for attacks and supports by sets of arguments. A set  $S$  is said to attack an argument  $a$  iff there is a  $T \subseteq S$  such that  $TR_a a$ .  $S$  is a minimal attack on  $a$  if there is no subset of  $T$  that also attacks  $a$ . We say that a set  $S$  attacks a set  $S'$  if  $S$  attacks one of the members of  $T$ . Similar notions exist for support, but are complicated by the fact that an argument is not supported by simply appearing in the  $R_e$  relation. In an evidence based approach, an argument is only acceptable if it is supported by another supported argument, or by evidence from the environment.

**Definition 2. (Evidential Support)** An argument  $a$  is e-supported by a set  $S$  iff

1.  $SR_e a$  where  $S = \{\eta\}$  or
2.  $\exists T \subset S$  such that  $TR_e a$  and  $\forall x \in T$ ,  $x$  is supported by  $S \setminus \{x\}$

$S$  is a minimum support for  $a$  if there is no  $T \subset S$  such that  $a$  is supported by  $T$ .

If  $a$  is e-supported by  $S$ , we may say that  $S$  e-supports  $a$ .

This notion of support is stronger than the one present in BAFs, and, in our opinion, offers a far stronger justification for the inclusion of a support relation within abstract argumentation systems. Our notion of support requires evidence at the start of a chain of support (i.e.  $\{\eta\}R_e x$  for some argument  $x \in S$ ) which leads, through various arguments to  $a$ , before the argument  $a$  may be used. With this notion, we may define the notion of an evidence-supported (or e-supported) attack:

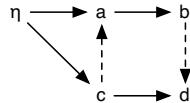
**Definition 3. (Evidence-Supported Attack)** A set  $S$  carries out an evidence-supported attack on an argument  $a$  if

- $X R_a a$  where  $X \subseteq S$ , and,
- All elements  $x \in X$  are supported by  $S$ .

A supported attack by a set  $S$  is minimal iff there is no  $T \subset S$  such that  $T$  carries out an evidence-supported attack on  $a$ .

An e-supported attack attempts to represent the notion of an attack backed up by evidence or facts, and is necessary for our definition of acceptability.

Support for an argument is clearly one requirement for acceptability. However it is not enough. Following Dung, an argument should be acceptable (with respect to some set) if it is defended from attack by that set. The question arises whether all attacks should



**Figure 1.** An evidential argument system with four arguments; dashed arrows represent attacks relations and solid arrows represent supports relations. Evidence nodes are those supported by the special argument  $\eta$ .

be defended against, or only e-supported attacks. In this work, we choose the latter, and, by doing so, we allow an argument to be defended from attack by having the attack itself attacked or by having some means of support for the argument attacked by the defending set.

**Definition 4. (Acceptability)** An argument  $a$  is acceptable with respect to a set  $S$  iff

1.  $S$  e-supports  $a$ , and
2. Given a minimal evidence-supported attack  $X \subseteq 2^A$  against  $a$ ,  $\exists T \in S$  such that  $TR_a x$ , where  $x \in X$  so that  $X \setminus \{x\}$  is no longer a supported attack on  $a$ .

An argument is thus acceptable with respect to a set of arguments  $S$  if any argument that e-support attacks it is itself attacked (either directly or by being rendered unsupported) by a member of  $S$ . The set  $S$  must also e-support the acceptable argument. It is clear from this definition of acceptability that an asymmetry arises with regards to acceptability, as we ignore elements of  $S$  that might themselves be attacked by other arguments, which would prevent  $S$  from supporting  $a$ . To overcome this problem, the notion of admissibility is required, and for this, we must first define two other notions.

**Definition 5. (Conflict free Sets)** A set of arguments  $S$  is conflict free iff  $\forall y \in S, \nexists X \subseteq S$  such that  $XR_a y$ .

**Definition 6. (Self Supporting Sets)** A set of arguments  $S$  is self supporting iff  $\forall x \in S, S$  e-supports  $x$ .

It should be noted that this definition of a conflict free set is very strong. A weaker definition is possible by defining a conflict free set as one not containing any e-supported attacks on itself. Since we intend to use the concept of a conflict free set in our definition of admissibility, we will show in Lemma 1 that, for our purposes, these two definitions of a conflict free set coincide.

It is clear (from Definition 2) that if a set is a minimum e-support for an argument, then the set is self supporting. We may thus define the concept of admissibility:

**Definition 7. (Admissible Set of Arguments)** A set of arguments  $S$  is said to be admissible iff

1. All elements of  $S$  are acceptable with respect to  $S$ .
2. The set  $S$  is conflict free.

Figure 1 illustrates these concepts. Here,  $a, b, c, d$  are all supported arguments. The set of arguments  $\{a, b\}$  e-support attacks  $d$ , while  $c$  e-support attacks  $a$ . Argument  $d$  is acceptable with respect to  $\{c\}$ , since  $c$  prevents  $b$  from being e-supported. Finally, the set  $\{c, d\}$  is an admissible set.

We may then show a number of useful results (proofs for the lemmas can be found in [7]):

**Lemma 1.** *An admissible set contains no supported attacks on itself.*

**Lemma 2.** *An admissible set is self supporting.*

**Lemma 3.** *The empty set is admissible.*

**Lemma 4.** *Given an admissible set  $S$ , and two arguments  $x, y$  which are acceptable with respect to  $S$ ,*

1.  $T = S \cup x$  is admissible,
2.  $y$  is acceptable with respect to  $T$ .

Given this groundwork, we may now define modified versions of a number of Dung's extensions.

**Definition 8. (Evidential Preferred Extensions)** *An admissible set  $S$  is an evidential preferred extension if it is maximal with respect to set inclusion. That is, there is no admissible set  $T$  such that  $S \subset T$ .*

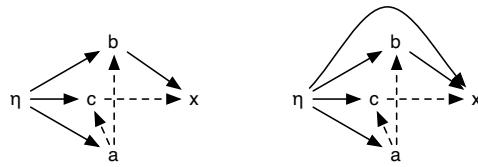
The evidential preferred (or e-preferred) extension is analogous to Dung's preferred extension, and we may prove the following result:

**Lemma 5.** *Any argumentation system has at least one e-preferred extension.*

It should be noted that Dung's preferred extension can be captured in our framework by having support exist only between the environment ( $\eta$ ) and all other arguments in the system.

There is no trivial transformation between BAF d/s/c-preferred semantics and our e-preferred semantics (or, as will be seen when other semantics for the evidential framework are described, between those and any BAF semantics). The reason for this is that the type of consistency BAF semantics describe is completely different to the consistency we are interested in. A set of arguments is deemed “compatible”, and thus part of an extension in one of the BAF semantics if, in some sense, none of the arguments in the extension both support, and attack some argument. Two types of compatibility were identified, namely internal, and external compatibility. Internal compatibility occurs when an argument is not accepted into the set if it would lead to the defeat of a member of the set. External compatibility on the other hand means that an argument would not be accepted into a set because its acceptance would mean that an argument outside the set would be both directly, or indirectly, supported and attacked by members of the set. We do not see this simultaneous attack and defence as a problem, additional evidence may allow us to still have a consistent set of arguments in this case (as shown by the semantics introduced here).

We may also introduce analogies to Dung's stable and grounded extensions. Informally, an e-stable extension is one that ensures that nothing but the e-stable set can be derived. This can be achieved by attacking arguments not in the set, or by attacking the support for those arguments.



**Figure 2.** Two sets of arguments and their interactions.

**Definition 9. *E-stable Extensions*** An *e-stable extension* of an argument framework  $A$  is a conflict free and self supporting set  $S$  such that for any supported argument  $a \notin S$ ,

1.  $S$  (support) attacks  $a$  or
2.  $\forall T$  such that  $T$  minimally supports  $a$ ,  $S$  (support) attacks  $T$ .

**Lemma 6.** A set  $S$  is an *e-stable extension* iff  $S = \{a | a \text{ is not support attacked by } S \text{ and is supported by } S\}$ , and  $S \neq \{\}\}$ .

To introduce the evidential grounded extension, we first introduce the characteristic function of an argument framework:

**Definition 10. (*The Characteristic Function*)** The characteristic function of an argumentation framework  $A$  is the function

$$F_A : 2^A \rightarrow 2^A$$

$$F_A(S) = \{a | a \text{ is acceptable with respect to } S\}$$

The following lemmas then hold:

**Lemma 7.** A conflict free set  $S$  of arguments is admissible iff  $S \subseteq F_A(S)$

**Lemma 8.**  $F_A$  is monotonic with respect to set inclusion.

From this, we can define the evidential grounded extension:

**Definition 11. (*E-grounded Extensions*)** An evidential grounded extension of a finitary argument framework  $A$  is the least fixed point of  $F_A$ .

#### 4. Discussion and Future Work

Evidential extensions are able to represent situations other frameworks are unable to handle. For example, consider the following scenarios:

1. The lefthand diagram in Figure 2 represents the following set of arguments: (a) it was dark, when it is dark, a witness statement could be wrong; (b) witness  $b$  said he saw a man fly by waving his arms. A witness statement can be viewed as evidence for a claim; (c) witness  $c$  said he saw a man waving his arms, but the man didn't fly. A witness' statement can be viewed as evidence for a claim; and (x) From the evidence, we can conclude that the man flew.

2. The righthand diagram in Figure 2 represents the following set of arguments: (a) It was dark, when it is dark, a witness statement could be wrong; (b) Witness  $b$  made the statement that the bird flew. A witness statement can be viewed as evidence; (c) Witness  $c$  made the statement that the bird did not fly. A witness statement can be viewed as evidence; and (x) We know that birds can normally fly, and thus given some evidence, we may claim that birds fly.

Given the arguments outlined in scenario 2, we would want our extension to contain the arguments  $\{a, x\}$ ; i.e. we would consider the arguments that it was dark and when it is dark a witness statement could be wrong, and we know birds can normally fly, and thus given some evidence we may claim that birds fly. In scenario 1, we would, intuitively, want to accept argument  $a$ ; the removal of some background knowledge (birds, by default, fly; humans do not) means that, unlike for the original argument  $x$ , we have no evidence one way or another that will allow us to determine the status of  $x$ .

The evidence-based argumentation framework presented in this paper neatly captures the distinction between these two cases; in scenario 1 the e-preferred extension consists of  $\{a\}$ , whereas in scenario 2 the e-preferred extension consists of  $\{a, x\}$ . Both extensions, therefore, correspond to our intuitions.

In contrast, it is not possible to capture this distinction within BAFs. This is simply because the notion of support within BAFs is not intended to capture the grounding of a chain of arguments in evidence. Support within BAFs captures a form of inference, which, as discussed in Section 2, should be captured at the level of logic, not argument. Our notion of *evidential support* from a *prima facie* argument is importantly distinct from the notion of support within BAFs, and, we believe, provides the first clear justification for including a notion of support within abstract argumentation frameworks.

Dung's extensions are clearly unable to handle the notion of support, and cannot represent either scenario, while the various extensions described by BAFs cannot distinguish between the two scenarios.

This added expressiveness is useful wherever reasoning with evidence, or with default and *prima facie* arguments, appears. For example, when performing contract monitoring, an agent could determine which clauses definitely hold by computing the e-grounded extension based on what it knows about the environment. In a contract enforcement setting, where two agents disagree as to what the contract state is, they could advance arguments, and the shared elements of the e-preferred extensions could again be used to determine which clauses are in force.

Our extensions reduce to Dung's original extensions when support links only exist between the empty argument and all other arguments. Since d-preferred extensions are identical to Dung's preferred extension, they can be similarly encapsulated by our framework. S-preferred and c-preferred extensions overlap with our extensions only for certain configurations of support.

As in BAFs, attack in our framework is more powerful than support. That is, if an attack succeeds, no amount of support will be able to negate it. One way of overcoming this weakness involves the concept of valuation of arguments [3], and we are actively pursuing this avenue of research.

Many relations between extensions are known to exist in Dung's framework as well as in BAFs. We intend to investigate which of these properties extend to our framework. Other extensions have been proposed for argumentation frameworks [2], and we intend to see how these translate into our more expressive model.

## 5. Conclusions

In this paper we have presented a novel abstract argumentation framework that captures the concepts of evidence-based support and support by multiple arguments. We have introduced a number of extensions allowing us to reason about sets of arguments in various ways. These extensions are based on the concepts of *evidence* and *support*; that is, an argument is only considered part of an extension if it is supported by a chain of arguments rooted by evidence, and is defended from attack by other arguments which are in turn supported by evidence. We have been able to prove many of the results shown by Dung for his original extensions, but are able to support more complicated arguments. Furthermore, our extensions are able to provide more intuitive results than Bipolar Argumentation Frameworks in cases where evidential reasoning is required; we would argue that this would include most, if not all, realistic domains. Finally, we believe that this research provides clear justification for the inclusion of the notion of support within abstract argumentation frameworks.

## 6. Acknowledgements

The first author is supported by the EU 6th Framework project CONTRACT (INFSO-IST-034418). The opinions expressed herein are those of the named authors only and should not be taken as necessarily representative of the opinion of the European Commission or CONTRACT project partners.

## References

- [1] L. Amgoud, C. Cayrol, and M.-C. Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *Proceedings of the 10th International Workshop on Non-monotonic Reasoning*, pages 1–9, Whistler, Canada, 2004.
- [2] C. Cayrol, C. Devred, and M.-C. Lagasquie-Schiex. Handling controversial arguments in bipolar argumentation systems. In *Proceedings of the 2006 Conference on Computational Models of Argument*, pages 261–272, 2006.
- [3] C. Cayrol and M.-C. Lagasquie-Schiex. Graduality in argumentation. *Journal of Artificial Intelligence Research*, 23:245–297, 2005.
- [4] C. Cayrol and M.-C. Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Pro. of the Eighth European Conference on Symbolic and Quantitative Approaches to Reasoning With Uncertainty*, volume 3571 of *LNAI*, pages 378–389. Springer-Verlag, 2005.
- [5] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
- [6] S. H. Nielsen and S. Parsons. A generalization of Dung’s abstract framework for argumentation: Arguing with sets of attacking arguments. In *Proceedings of the Third International Workshop on Argumentation in Multi-Agent Systems*, pages 7–19, Hakodate, Japan, 2006.
- [7] N. Oren. *An Argumentation Framework Supporting Evidential Reasoning with Applications to Contract Monitoring*. Phd thesis, University of Aberdeen, Aberdeen, Scotland, 2007.

# Argument Schemes and Critical Questions for Decision Aiding Process

Wassila OUERDANE<sup>a,1</sup>, Nicolas MAUDET<sup>a</sup> and Alexis TSOUKIAS<sup>a</sup>

<sup>a</sup> LAMSADE, Uni. Paris-Dauphine, Paris 75775

**Abstract.** Our ambition in this paper is to begin to specify in argumentative terms (some of) the steps involved in a decision-aiding process. To do that, we make use of the popular notion of argument schemes, and specify the related critical questions. A hierarchical structure of argument schemes allows to decompose the process into several distinct steps—and for each of them the underlying premises are made explicit, which allows in turn to identify how these steps can be dialectically defeated *via* critical questions. This work initiates a systematic study which aims at constituting a significant step forward for forthcoming decision-aiding tools. The kind of system that we foresee and sketch here would allow: (i) to present a recommendation that can be explicitly justified; (ii) to revise any piece of reasoning involved in this process, and be informed of the consequences of such moves; and possibly (iii) to stimulate the client by generating contradictory arguments.

**Keywords.** Decision aiding, argument schemes, critical questions

## Introduction

Decision theory and multiple criteria decision analysis have established the theoretical foundations upon which many decision-support systems have blossomed. However, such systems have focussed more on how a “best solution” should be established, and less on how a decision maker should be convinced about that (for exceptions on that see [9,5]). In addition, the decision-support process is often constructive, in the sense that the client refines its formulation of the problem when confronted to potential solutions. This requires the system to cater for revision: it should be possible, for the client, to refine, or even contradict, a given recommendation. These aspects are usually handled by the decision analyst, but if we are to automate (some part of) the process (as is the case in recommender systems, for instance), it is important to understand more clearly how they can be integrated in a tool.

In AI, a different tradition to decision making had identified these problematic issues. One of the key distinctive ingredient is that many AI-based approaches are prone to represent decision making in terms of “cognitive attitudes” (as exemplified in the famous Belief-Desire-Intention paradigm) [11,12], instead of crude utilities (as already elicited by the analyst). This change of perspective paved the way for more flexible decision-making models: goals may change with circumstances, and understanding these under-

---

<sup>1</sup>Corresponding Author.

lying goals offers the opportunity to propose alternative actions, for example. But then a reasoning machinery has to be proposed to handle these complex notions. Regarding the issues of expressiveness and ability to deal with contradiction that we emphasized here, argumentation seemed a good candidate. Indeed, recently, following some early works [13,8], several models have been put forward in the artificial intelligence community that make use of argumentation techniques [16] to attack decision problems. These approaches have contributed to greatly extend our understanding of the subject, in particular they clarify what makes argumentation about actions crucially different from mere epistemic argumentation (when the object under discussion is a belief).

On the one hand, our contribution is much more modest in its current state than the aforementioned approaches. We will not, in the present paper, base our model on cognitive attitudes and try to represent the underlying motivations and informations of agents. We take, instead, a different perspective which results from the following observation: there exist many decision-support tools that clients understand well, find valuable, and would be reluctant to drop for a completely new tool. Hence the following question: is it possible (and to what extent) to integrate some flavour of argumentation within these tools. On the other hand, having to deal with complex aggregation procedures proposed in these approaches, we will also have to make explicit and discuss some aspects that are often left aside by argumentation-based approaches (although it is known that some aggregation procedures can be captured by an argumentative approach [1]). The main one being that the aggregation procedure itself may be the subject of potential exchange of arguments.

The remainder of this paper is as follows. Section 1 offers a brief reminder on decision aiding theory. In particular we identify the different steps that compose the process, and the nature of the involved objects. Section 2 then presents the different argument schemes that are involved in such processes. The section that follows exploits this representation and defines the critical questions that can be attached to the argument schemes. In Section 4, we present the nature of the resulting dialectical process, pointing out the added-value of this argumentation-based approach when compared to classical multicriteria decision-aiding tools. We conclude by discussing perspectives of this work.

## 1. Decision Aiding Process

An instance of a decision process is characterized by the participating actors, their concerns, and the resources committed by each actors on each object. We are interested in decision aiding. Intuitively, in decision aiding we also make decisions (what, why and how to model and support). Decision aiding is also a decision process but of a particular nature [9,10]. A decision aiding context implies the existence of at least two distinct actors (the client and the analyst) both playing different roles; at least two objects, the client's concern and the analyst's (economic, scientific or other) interest to contribute; and a set of resources including the client's domain knowledge, the analyst's methodological knowledge, money, time... The ultimate objective of this process is to come up with a consensus between the client and the analyst [19]. Four cognitive artifacts constitute the overall process:

- *Problem situation*— the first deliverable consists in offering a representation of the problem situation for which the client has asked the analyst to intervene;

- *Problem formulation*—given a representation of the problem situation, the analyst may provide the client with one or more problem formulation. The idea is that a problem formulation translates the client's concern, using the decision support language, into a “formal problem”;
- *Evaluation Model*—for a given problem formulation, the analyst may construct an evaluation model, that is to organise the available information in such a way that it will be possible to obtain a formal answer to a problem statement. An evaluation model can be viewed as a tuple comprising the set of alternatives on which the model applies (denoted  $\mathcal{A}$ ); the set of dimensions (attributes) under which the elements of  $\mathcal{A}$  are observed, described, measured, etc. (denoted  $\mathcal{D}$ ); the set of scales  $\mathcal{E}$  associated to each element of  $\mathcal{D}$ ; the set of criteria  $\mathcal{H}$  under which each element of  $\mathcal{A}$  is evaluated in order to take in account the client's preference; and an aggregation procedure ( $\mathcal{R}$ ). Formally, a criterion is a preference relation, that is a binary relation on  $\mathcal{A}$  or a function representing the criterion. (A set of uncertainty structures may also be used. Depending on the language adopted, this set collects all uncertainty distributions or the beliefs expressed by the client. We shall not discuss it further here).
- *Final recommendation*—the evaluation model will provide an output which is still expressed in terms of the decision support language. The final recommendation is the final deliverable which translate the output into the client's language.

The study of this process shows that it suffers from some limits. The first one is the lack of a formal justification or explanation of the final recommendation. Indeed, the process focuses more on how to reach the final decision and fails in some way to provide a justification for the decision-maker. Second, during the decision aiding process several different versions of the cognitive artifacts may be established. These different versions are due to the fact that client doesn't known how to express clearly, at the beginning of the process, what is his problem and what are his preferences. So, as the model is constructed, the decision maker revise and update his preferences and/or objectives. However, such different versions are strongly related to each other since they carry essentially the same information and only a small part of the model has to be revised [19,10]. The problem that arises here is that this revision (or update) is not taken into account by the model. In other words, there is no formal representation of how the evolution occurs between different versions. Finally, the last problem encountered in this process is the incomplete information. More specifically, the process does not support situations or problems decision where some fields of one or more of the different models are not completed.

In this paper we concentrate on the evaluation step. The approach based on argumentation that we sketch in the next few sections is particularly well suited to tackle these aspects: (i) by presenting the reasoning steps under the form of argument schemes, it makes justification possible, and offers the possibility to handle default reasoning with incomplete models; and (ii) by defining the set of attached critical questions, it establishes how the revision procedure can be handled.

## 2. Argument Schemes

Argument schemes are argument forms that represent inferential structures of arguments used in everyday discourse, and in special contexts like legal argumentation, or scientific

argumentation. *disjunctive syllogism* are very familiar. But some of the most common and interesting argumentation schemes are neither deductive nor inductive, but *defeasible and presumptive* [22].

It is now well established that argument schemes can play two roles: (i) when constructing arguments, they provide a repertory of forms of argument to be considered, and a template prompting for the pieces that are needed; (ii) when attacking, arguments provide a set of critical question that can identify potential weaknesses in the opponents case. Then, as Walton puts it, “ we have two devices, *schemes* and *critical questions*, which work together. The first device is used to identify the premises and conclusion. The second one is used to evaluate the argument by probing into its potentially weak points” [22]. The set of critical questions have to be answered, when assessing whether their application in a specific case is warranted. Prakken and Bench-capon [6] specify that argument schemes are not classified according to their logical form but according to their content. Some argument schemes express epistemological principles or principles of practical reasoning: different domains may have different sets of such principles. Our aim in this paper to identify those schemes that are involved in multicriteria decision-aiding processes.

We need different classes of argument schemes to construct the whole evaluation model. Argument schemes can very broadly be distinguished depending on (i) whether they aggregate several criteria, or are concerned with a single criteria (multicriteria vs. unicriteria); (ii) whether they follow a pairwise comparison principle or whether they use an intrinsic evaluation, the action being compared to a separation profile (intrinsic vs. pairwise); and (iii) whether they are concerned with the evaluation of the action or its mere acceptability (evaluation vs. acceptability). In theory, all combinations seem possible, even though some are much more natural than others.

In this paper, we shall focus our attention on the following schemes:

- argument schemes for *Unicriteria Pairwise Evaluation* (UC-PW-EV), which establishes that an objet is at least prefered to another object from the single viewpoint of the considered criteria (note that there may be an intrinsic version of this scheme, for instance for classification, but also to cater for all the argumentation-based aggregation techniques);
- argument schemes for *Unicriteria Intrinsic Acceptability* (UC-IN-AC), which establishes that the action can be considered in the evaluation process (here also, it may be possible to have a similar scheme for relative “pairwise” acceptability);
- argument scheme for *Multicriteria Pairwise Evaluation* (MC-PW-EV), which basically concludes that an object is at least as good as another object on the basis of several criteria taken together. It is constituted of two sub-argument schemes:
  - \* argument schemes for *Positive Reasons Aggregation Process* (PR-AP), which concludes that there are enough positive reasons to support the claim of MC-PW-EV, and that can be of many types depending on the aggregation technique used (ex. simple majority, weighted sum, and so on);
  - \* argument schemes for *Negative Reasons Aggregation Process* (NR-AP) which concludes that the negative reasons should block the conclusion of MC-PW-EV (again, this really constitute a family of argument scheme);

- argument schemes for *Global Recommendation* (GR) which provides the output of the process (of different type depending on the decision problem considered). We shall not discuss this level in this paper.

In the rest of this paper we limit the discussion to the case involving only two actions. This is a basic building block that will be required if we are to construct more general decision-aiding.

Now we turn our attention to argument schemes. In fact, as must be clear from the discussion above, there is an underlying hierarchical structure that ties the different argument schemes. In short, we can distinguish three levels of argument schemes that will be embedded. At the highest level the multicriteria pairwise evaluation, which is based on the aggregation of positive and negative reasons, which in turn is based on unicriteria evaluation of actions versus other actions (or special profiles).

### 2.1. Argument Schemes for Unicriteria Action Evaluation

The first way to perform an action evaluation is to compare two actions from the point of view of the chosen criterion: this is modeled by the scheme for Unicriteria Pairwise Evaluation (UC-PW-EV), see Tab. 1. This argument scheme is the basic piece of reasoning that is required in our decision-aiding context. It concludes that an action  $a$  is at least as good as an action  $b$  from the point of view of a given criterion  $h_i$ , based on some preference relation  $\succeq_i$  [17].

<b>Premises</b>	a criteria	$h_i$
	an action	$a$
	whose performance is	$g_i(a)$
	an action	$b$
	whose performance is	$g_i(b)$
	a preference relation	$\succeq_i$
<b>Conclusion</b>	$a$ is at least as good as $b$	$a \succeq_i b$

**Table 1.** Scheme for Unicriteria Pairwise evaluation (UC-PW-EV)

When an action needs to be intrinsically evaluated, there is a need to define the categories and *separation profiles*. Such a separation profile defines on each criterion a sort of neutral point: this is by not necessarily an existing action, but it allows to define to which category to affect the action. A particular case is when we only consider “pro” and “con” categories. The scheme for *Unicriteria Intrinsic Action Evaluation*, as given in Tab. 2, details such a scheme.

<b>Premises</b>	an action	$a$
	whose performance is	$g_i(a)$
	along a criteria	$h_i$
	a separation profile	$p$
	whose performance is	$g_i(p)$
	a preference relation	$\succeq_i$
<b>Conclusion</b>	$a$ is acceptable according to $h_i$	$a \succeq_i p$

**Table 2.** Scheme for Unicriteria Intrinsic Action Evaluation (UC-IN-EV)

## 2.2. Argument Schemes for Acceptability

The case of action acceptability is very similar to that of action evaluation: it can also be performed intrinsically or in pairwise manner. We start with the *Argument Scheme for Intrinsic Acceptability* (UC-IN-AC). The scheme is very similar to that of Unicriteria Intrinsic Evaluation. In fact, in this case the separation profile can play the role of a *veto threshold*: when the action does not reach that point, there are good reasons to exceptionally block the claim (disregarding the performance of the action on other criterion). For the sake of readability, we shall not repeat this very similar scheme here. A different kind of acceptability relies instead on the relative comparison of actions: it may be the case that an action is considered to be unacceptable because the difference in performance is so huge with another action. In this case, we talk about an *Argument Scheme for Pairwise Acceptability* (UC-PW-AC). We believe this is self-explanatory given the examples provided so far, and shall not give any further detail here.

## 2.3. Arguments Scheme for Aggregating Positive Reasons

At this level the piece of reasoning involved must make clear how we can conclude that enough positive reasons are provided. Perhaps the most obvious such scheme, at least one that is ubiquitous in multicriteria making is the *principle of majority*. It only says that  $a$  is at least as good as  $b$  when there is a majority of criterion supporting this claim. Table 3 gives the detail of the corresponding argument scheme.

<b>Premises</b>	a set of criteria considered to be of equal importance a set of pairwise evaluation of actions $a$ and $b$ the majority support the claim	$\{h_1, h_2, \dots, h_n\}$
<b>Conclusion</b>	there are good reasons to support $a$ is at least as good as $b$	$a \succeq b$

Table 3. Scheme for Argument from the Majority Principle (PR-AG (maj))

Note that this scheme makes explicit that criteria are considered to be of equal importance. This is not necessarily the case, and more generally many other aggregation techniques may be used to instantiate  $\mathcal{R}_P$ . These other schemes will potentially require additional information, which justifies that we have many different scheme and not a single generic one. For instance, a possible scheme would conclude that  $a$  is at least as good as  $b$  when it is at least as good on (some of) the most important criteria (*argument from sufficient coalition of criteria*).

Here we only present a different one to illustrate the variety of argument schemes that may be used. This simple typical example is the lexicographic method that we detail below. The method works as follow: look at the first criterion, if  $a$  is strictly better than  $b$  on this, then  $a$  is declared globally preferred to  $b$  without even considering the following criteria. But if  $a$  and  $b$  are indifferent on the first criterion, you look at the second one, and so on.

Note that the basic input information that needs to be provided to these schemes is that of a pairwise comparison on a single criterion dimension (the output of UC-PW-EV). Indeed, this will be in most case the basic building block upon which the recommendation can be build. There is however a different type of scheme that would aggregate instead intrinsic valuations of both actions: that would be the case of argument-based

<b>Premises</b>	a set of criteria	$\{h_1, h_2, \dots, h_n\}$
	a linear order on the set of criteria	$h_1 > h_2 > \dots > h_n$
	a set of pairwise evaluation of actions $a$ and $b$	
	$a$ is strictly better than $b$ on $h_i$	$a \succ_i b$
	$a$ is indifferent to $b$ on $h_j$ for any $j < i$	$a \simeq_j b$ when $j < i$
<b>Conclusion</b>	there are good reasons to support $a$ is at least as good as $b$	$a \succeq b$

**Table 4.** Scheme for Argument from the lexicographic method (PR-AG (lex))

aggregation procedures that take as input sets of arguments “pro” and “con”. Clearly, the basic argument scheme required will be different here, for it needs to provide an intrinsic evaluation of the action.

#### 2.4. Argument Scheme for Multi-Criteria Pairwise Evaluation

The argument scheme that lies at the top of our hierarchy is inspired by outranking multi-criteria techniques [10], and indeed its argumentative flavour is obvious. The claim holds when enough supportive reasons can be provided, and when no exceptionally strong negative reason is known. This already suggests that there will be (at least) two ways to attack this argument: either on the basis on a lack of positive support, or on the basis of the presence of strong negative reasons (for instance, a “veto”). Typically, supportive reasons are provided by action evaluation, and negative reasons are provided by action (lack of) acceptability. We shall discuss this further when we turn our attention to critical questions.

<b>Premises</b>	an action	$a$
	an action	$b$
	a set of criteria	$\{h_1, h_2, \dots, h_n\}$
	there are enough supportive reasons according to	$\mathcal{R}_P$
	there are no sufficiently strong reasons to oppose it	$\mathcal{R}_N$
<b>Conclusion</b>	$a$ is at least as good as $b$	$a \succeq b$

**Table 5.** Scheme for pairwise evaluation multicriteria (MC-PW-EV)

Here,  $\mathcal{R}_P$  stands for the aggregation process that should be used to aggregate the (positive) reasons supporting the claim, whereas  $\mathcal{R}_N$  stand for the aggregation process concerned with the aggregation of *exceptionally* negative reasons (vetos). The conclusion of the scheme expresses that  $a$  is at least as good as  $b$  according to the preference relation  $\succeq_{MC\,PW\,EV}$  induced by the scheme.

### 3. Critical Questions

Along with each different argument schemes comes a set of *critical questions* [22,21]. These questions as we said before, allow us to identify potential weaknesses in the scheme. Below we present the set of critical questions attached to the schemes MC-PW-EV, PR-AG (maj), and UC-PW-EV. We note that different types of critical questions can be identified [14], depending on whether they refer to standard assumptions of the scheme or to exceptional circumstances. This has in particular a significant difference on how the burden of proof is allocated. We now list some of the questions that can be attached to the different premises.

*Argument Scheme for Multi-Criteria Pairwise Evaluation.* In this context the different type of questions is clear. The burden of proof lies on the proponent when it must provide supportive evidence (positive reasons) for the main claim. On the other hand, the opponent should be the one providing negative reasons to block the conclusion.

1. *actions* (assumption): is the action possible?
2. *list of criteria* (assumption): (i) Is this criteria relevant?, (ii) Should we introduce a new criteria?, (iii) Are these two criteria are in fact the same?
3. *positive reasons* (assumption): (i) Are there enough positive reasons to support the claim? (ii) Is the aggregation technique relevant ?
4. *negative reasons* (exception): Are there not enough reasons to block the claim?  
Is the aggregation technique relevant?

Note also that while the use of a specific aggregation technique may be challenged at this level (“why are we using a majority principle here?”), the actual exchange of argument regarding this aspect will involve the sub-argument scheme concerned with this aggregation. We now turn our attention to the critical questions that may then be used.

Together with the *Scheme for Argument from the Majority Principle* come two obvious questions:

1. *list of criteria* (exception): Are the criteria of equal importance?
2. *majority aggregation* (exception): Is the simple majority threshold relevant for the current decision problem?

As for the *Argument Scheme for Unicriteria Pairwise Action Evaluation*, we can propose this tentative set of questions :

1. *actions* (assumption): Is the action possible?
2. *criterion* (assumption): Is the criteria relevant?
3. *action's performance* (assumption): Is the performance correct?
4. *preference relation* (assumption): Is the preference relation appropriate?

It should be noted that a negative answer to some of these questions leads to a conflict whose resolution requires sometimes the transition to a different stage of the negotiation process. For instance, when you challenge whether the action is possible to start with, you are dealing with the problem formulation (cf. section 1), where the set of alternatives is defined. It is out of the scope of this paper to discuss this problem. We will just mention that through the different critical questions, we have the opportunity to review and correct not only the evaluation model, but also other stages of the process.

#### 4. The Dialectical Process

In this section we give a glimpse of the dialectical process that will exploit the argument schemes and critical questions that we have put forward so far. It is based on the popular model of dialogue games, and more precisely it is based on recent extensions that incorporate argument schemes within such models [18]. The full specification of the dialogue game is the subject of ongoing work. The process initiates with the client specifying the

basic elements of the evaluation model<sup>2</sup> (see Sect. 1): it specifies a set of actions (in the context of this paper we limit ourselves to two actions though), a set of criteria, and the aggregation operators that shall be used. Contrary to classical decision tools, these sets will only be considered to be the *current* evaluation model, and it is taken for granted that it can be revised throughout the process. Now, as we see it, an argumentation-based decision-aiding process should:

1. *justify* its recommendation. Crucially, by presenting its justifications in the form of arguments, the system will make it possible for the user to pinpoint those steps that pose problems. The system builds up the current recommendation by embedding argument schemes of the three levels. The argument schemes are build on the basis of the information provided by the user, and in some cases by using default instantiation (when the scheme allows for it). If challenged by the user, the system provides the different steps of reasoning by revealing parsimoniously the lower level schemes that compose the recommendation. Each time a scheme is presented, the entire set of critical is at the disposal of the user to challenge the current conclusion. There are very different reasons to revise in such a process: in some case the user may simply want to correct/refine one of its previous statement, introduce new information. In other cases it will contradict one of the system's assumption;
2. *inform* the user of the consequences of its modifications. In many cases the user would not foresee the various consequences of a seemingly local modification: in these cases, the system helps the user by making explicit the hidden and critical consequences of its move;
3. potentially *propose* automatically generated arguments, in order to stimulate users (a technique that has proven very useful, for instance in the context of recommender systems [20]). The system holds a collection of argument schemes that can be readily used to generate counter-arguments that would stimulate the user in its decision-making process. In particular, the system may seek to generate counter-arguments based on the use of different argument schemes that would contradict the current recommendation, by relaxing some assumptions or informations provided by the user. The idea being here to emphasize for the user what are the assumptions in the evaluation model currently used.

The different cases mentioned above are illustrated by means of an example.

*User: [specify its evaluation model: two actions a and b, five criteria c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>, c<sub>4</sub>, c<sub>5</sub>, among which c<sub>1</sub> is considered to be of primary importance, followed by c<sub>2</sub>, and so on. Acceptability is initially intrinsic: no action should have a score on a criterion lower than 1. The performance table is as follows:]*

	c <sub>1</sub>	c <sub>2</sub>	c <sub>3</sub>	c <sub>4</sub>	c <sub>5</sub>
a	7.5	6	2	3	5
b	7	4	8	4	7

1. *System: Given your informations, a is at least as good as b. [Recommendation]*
2. *User: Why? [Challenge]*

---

<sup>2</sup>Of course a more ambitious dialectical system would have to consider the previous steps as well. This is beyond the scope of this paper.

3. System: *The most important criteria according to you defend this claim, so by comparing actions on the basis of criteria of decreasing importance, a should be preferred to b [Justified Recommendation]*

*(although the majority of arguments defend the opposite claim [Gen. counter-arguments])*

4. User: *I still think that  $c_1$  is the most important criteria so the majority makes little sense. [Reject counter-argument]*

*however, now I see that the difference between a and b is not significant enough...*

5. System: *My recommendation remains the same because a is preferred to b on the second most important criterion*

6. User: *I agree but still a cannot be acceptable.*

7. System: *Why? [Burden of proof shift]*

8. User: *b is way better than a on  $c_3$ , I could not justify this choice... [Argument Scheme for Relative acceptability]*

9. System: *Ok. [Revise the recommendation]*

*But beware that applying this scheme for acceptability may have the following consequences... [Inform of consequences of this revision]*

Let us briefly analyze this dialogue. Turn 1 provides the recommendation, which is challenged by the user on turn 2. Not being more explicit, the challenge can be assumed to ask the system to provide more explicit information regarding positive reasons supporting the claim. The system, on turn 3, explains that the claim is based on the use of the PR-AG(lex) scheme. Note that it also generates a possible counter-argument by relaxing some of the information provided by the user (here the fact that criteria have different importance). Observe that this is an indirect way for the system to use a critical question...) The user rejects this counter-argument on turn 4 (by re-affirming the fact that criteria have unequal importance), but attacks the basic UC-PW-EV argument upon which the recommendation is based. The critical question used here is that of the relevance of the preference relation. The system accepts the move (and modifies the user's information by specifying that actions should exhibit at least half a point of difference, otherwise they should be considered as indifferent). But the system restates that the recommendation remains unchanged: this is due to the fact on the second most important criterion,  $a$  is again better than  $b$ . (The attack is *unrelevant* in Prakken's sense). The user accepts this but now attacks on the ground of negative reasons, and explains that  $a$  can not be accepted on the basis of pairwise acceptability (UC-PW-AC). Finally, the system revises its recommendation but may at the same time make explicit the consequences of the proposed change.

## 5. Related work

One of the most convincing proposal recently put forward to account for argument-based decision-making is the one by Atkinson *et al.* [3,2]. They propose an extension of the “sufficient condition” argument scheme for practical reasoning [21], by distinguishing the goal into three elements: state, goal and value. This scheme serves as a basis for the construction of a protocol for a dialogue game, called Action Persuasion Protocol (PARMA) [4]. The authors show how their proposal can be made computational within the framework of agents based on the BDI model, and illustrate this proposal with an

example debate within a multi-agent system. Prakken et al. [7] offer a logical formalisation of Atkinson's account within a logic for defeasible argumentation. They address the problem of practical syllogism by trying to answer questions such as: how can an action be justified? In particular, the aim is to take into account the abductive nature of the practical reasoning and the side effects of an action. A key element in this formalisation is the use of accrual mechanism for argument to deal with side effects (positive and negative effects).

The first approach attempting to introduce argumentation in the decision aiding process as a whole is the one of Moraitis et al. in [15]. The idea is to describe the outcomes of the decision aiding process through an operational model and to use argumentation in order to take into account the defeasible character of the outcomes. The authors try to provide a way allowing the revision and the update of the cognitive artifacts of the Decision Aiding Process.

In addition to these works, many other proposals have been put forward in the literature to use argumentation in a decision context, see [16] for a recent survey. From the point of *decision aiding* though, a couple of elements remain largely unexplored. Under that perspective, current argumentation models are not fully satisfying because for instance: (i) most of the approaches assume a decision problem where the aim is to select the “best” action for a given purpose, when in fact a variety of decision problems can be addressed (choice, ranking, sorting,...); and (ii) most models currently proposed in the literature rely on an underlying *intrinsic evaluation* (actions are evaluated against some absolute scale), whereas most decision aggregation procedure make use of *pairwise evaluation* techniques (actions are compared against each others).

## 6. Conclusion and Future Work

The purpose of this paper was to provide a first approach to represent the steps of a multicriteria decision aiding process by means of argument schemes and critical questions. We focused here on the evaluation model, and considered the restricting but basic case of the comparison of two actions. To represent the decision evaluation process, we identified a hierarchical structure of argument schemes. Each level refers to one step in the classical multicriteria evaluation. The highest level represents the pairwise evaluation, which is based on the aggregation level, which is in turn based on unicriteria evaluation (pairwise or intrinsic). To these schemes we associated a set of critical questions. One reviewer of this paper raised the following issue: does it make sense in the first place to consider argument schemes that cover the aggregation level? One of the main claim of this paper is that it does, precisely because the way basic argument schemes are collected and aggregated may also be disputed, and be based on assumptions that can be challenged and/or revised. The aim is (as usual with argument schemes and critical questions, as proposed here) to allow us to check the acceptability of each scheme by probing into its potentially weak points, and this from different point of views. We also give the very basic ingredients of the dialectical system currently under development. Future work should extend the model to take into account, in one hand a large set of alternatives, on other hand to handle different decision problems (ranking, sorting,...), in order to build a dialectical system-based decision aiding system for the whole process.

## References

- [1] L. Amgoud, J.-F. Bonnefon, and H. Prade. An Argumentation-based Approach to Multiple Criteria Decision . In *Proc. of the 8th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 269–280. LNCS, 2005.
- [2] K. Atkinson. Value-based argumentation for democratic support. In *Proc. of the 1st International Conf. on Computational Models of Natural Argument*, pages 47–58. IOS Press, 2006.
- [3] K. Atkinson, T. J. M. Bench-Capon, and S. Modgil. Argumentation for decision support. In *Proc. of the 17th International Conf. on Database and Expert Systems Applications*, pages 822–831, 2006.
- [4] K. Atkinson, T.J.M. Bench-Capon, and P. McBurney. Computational representation of practical argument. *Knowledge, Rationality and Action*, 152(2):157–206, 2006.
- [5] V. Belton and T. Stewart. *Multiple Criteria Decision Analysis: An Integrated Approach*. Kluwer Academic, Dordrecht, 2002.
- [6] T.J.M Bench-Capon and H. Prakken. Argumentation. In A.R. Lodder & A. Oskamp, editor, *Information Technology & Lawyers : Advanced technology in the legal domain from challenges to dailyroutine*, pages 61–80. Springer Verlag, Berlin, 2005.
- [7] T.J.M. Bench-Capon and H. Prakken. Justifying actions by accruing arguments. In P.E. Dunne and T.J.M. Bench-Capon, editors, *Proc. of the 1st International Conf. on Computational Models of Natural Argument(COMMA'06)*, volume 144, pages 311–322, Amsterdam, The Netherlands, 2006. IOS Press.
- [8] B. Bonet and H. Geffner. Arguing for Decisions: A Qualitative Model of Decision Making. In *Proc. of the 12th Conference on Uncertainty in Artificial Intelligence (UAI'96)*, pages 98–105, 1996.
- [9] D. Bouyssou, T. Marchant, M. Pirlot, P. Perny, A. Tsoukiàs, and Ph. Vincke. *Evaluation and decision models: a critical perspective*. Kluwer Academic, Dordrecht, 2000.
- [10] D. Bouyssou, T. Marchant, M. Pirlot, A. Tsoukiàs, and P. Vincke. *Evaluation and decision models with multiple criteria: Stepping stones for the analyst*, volume 86 of *International Series in Operations Research and Management Science*. Springer, Boston, 2006.
- [11] M. Dastani, J. Hulstijn, and L. van der Torre. How to decide what to do? *European Journal of Operations Research*, 160(3):762–784, 2005.
- [12] J. Doyle and R. Thomason. Background to qualitative decision theory. *AI magazine*, 20(2):55–68, 1999.
- [13] J. Fox and S. Parsons. On Using Arguments for Reasoning about Actions and Values. In *Proc. of the AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning*, pages 55–63. AAAI Press, 1997.
- [14] T. Gordon, H. Prakken, and D. Walton. The carneades model of argument and burden of proof. *Artificial Intelligence*, 171:875–896, 2007.
- [15] P. Moraitis and A. Tsoukiàs. Decision aiding and argumentation. *Proc. of the 1st European Workshop on Multi-Agent Systems*, 2003.
- [16] W. Ouerdane, N. Maudet, and A. Tsoukiàs. Arguing over actions that involve multiple criteria: A critical review. In *Proc. of the 9th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 308–319, 2007.
- [17] M. Oztürk, A. Tsoukiàs, and Ph. Vincke. Preference modelling. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 27–72. Springer Verlag, Boston, Dordrecht, London, 2005.
- [18] C. Reed and D. Walton. Argument schemes in dialogue. In H.V. Hansen, C.W. Tindale, R.H. Johnson, and J.A. Blair, editors, *Dissensus and the search for common ground*, 2007.
- [19] A. Tsoukiàs. On the concept of decision aiding process. *Annals of Operations Research*, 154(1):3–27, 2007.
- [20] P. Viappiani, B. Faltings, and P. Pu. Preference-based search using example-critiquing with suggestions. *Journal of Artificial Intelligence Research*, 171:465–503, 2006.
- [21] D.N. Walton. *Argumentation schemes for Presumptive Reasoning*. N. J., Erlbaum, 1996.
- [22] D.N. Walton and C.A. Reed. Argumentation schemes and defeasible inferences. In Giuseppe Carenini, Florina Grasso, and Chris Reed, editors, *Workshop on Computational Models of Natural Argument*, 2002.

# Arguments in OWL: A Progress Report

Iyad RAHWAN<sup>a,b,1</sup>, Bita BANIHASHEMI<sup>a</sup>

<sup>a</sup> Faculty of Informatics, The British University in Dubai, Dubai, UAE

<sup>b</sup> (Fellow) School of Informatics, University of Edinburgh, UK

**Abstract.** In previous work, we presented an RDFS ontology, based on the Argument Interchange Format (AIF), for describing arguments and argument schemes. We also implemented a pilot Web-based system, called ArgDF, for authoring and querying argument structures represented in RDF. In this paper, we discuss some of the limitations of our earlier reification of the AIF. We then present a new ontology which takes advantage of the higher expressive power of OWL. We demonstrate how this enables the use of automated Description Logic reasoning over argument structures. In particular, OWL reasoning enables significantly enhanced querying of arguments through automatic scheme classifications, instance classification, and inference of indirect support in chained argument structures.

**Keywords.** Argumentation, Argument Interchange Format, Semantic Web, OWL

## 1. Introduction

A number of Web 2.0 tools now provide explicit support for argumentation, enabling a more explicit structuring of arguments. Such tools include *Truthmapping*,<sup>2</sup> *Debatabase*,<sup>3</sup> *Standpoint*,<sup>4</sup> and *Standpedia*.<sup>5</sup> These systems have a number of limitations. There is limited or no integration between argument repositories. This limits the ability to provide services (e.g. question answering systems) that make use of arguments from multiple repositories, or the ability of users to easily access arguments across tools.

Another, related limitation of existing systems is that argument structure is relatively shallow. Most Web 2.0 applications distinguish only between premises and conclusions, and possibly between pro- and con- arguments. But they do not distinguish between different types of arguments, or subtle types of attack among arguments. Moreover, existing tools do not provide semantically rich links among arguments. For example, in truthmapping, while user-contributed text (i.e. premises, conclusions, critiques and rebuttals) can contain hyperlinks to any Web content including other arguments, which does enable cross-referencing among arguments, these references carry no explicit semantics (e.g. expressing that a link represents a support or an attack). This limits the possibilities for automated search and evaluation of arguments.

---

<sup>1</sup>Correspondence to: Iyad Rahwan, the British University in Dubai, P.O.Box 502216, Dubai, UAE. Tel.: +971 4 367 1959; Fax: +971 4 366 4698; E-mail: irahwan@acm.org.

<sup>2</sup><http://www.truthmapping.com>

<sup>3</sup><http://www.idebate.org/debatabase/>

<sup>4</sup><http://www.standpoint.com>

<sup>5</sup><http://www.standpedia.com>

Semantic Web technologies [1] are well placed to facilitate the integration among mass argumentation tools. A unified argument description ontology could act as an interlingua between the different tools. If Web 2.0 mass argumentation tools can provide access to their content through a common ontology, developers could build tools to exchange (e.g. import and export) or integrate arguments between tools. Another benefit of specifying arguments in standard ontology languages is the potential for automated inference over argument structures, such as inference based on Description Logic [2]. In previous work [3], we presented the first (pilot) realisation of a Semantic Web system for argument annotation, based on the argument interchange format (AIF) [4].

In this paper, we discuss some of the limitations of our earlier AIF reifications. We then present a new ontology which takes advantage of the higher expressive power of OWL [5]. We demonstrate how this enables the use of automated Description Logic reasoning over argument structures. In particular, OWL reasoning enables significantly enhanced querying of arguments through automatic scheme classifications, instance classification, and inference of indirect support in chained argument structures.

The paper advances the state of the art in the computational modelling of argumentation in two main ways. Firstly, the new OWL ontology significantly enhances our previous RDF Schema-based implementation [3].<sup>6</sup> In particular, we provide a new reification of the AIF specification and model schemes as classes (as opposed to instances), which enables explicit classification of schemes themselves. Secondly, our new system enables the first explicit use of Description Logic-based OWL reasoning for classifying arguments and schemes in a Web-based system. This provides a seed for further work that combines traditional argument-based reasoning techniques [7] with ontological reasoning in a Semantic Web environment.

## 2. Background: The Core Argument Interchange Format

The AIF is a core ontology of argument-related concepts, and can be extended to capture a variety of argumentation formalisms and schemes. The AIF core ontology assumes that argument entities can be represented as nodes in a directed graph called an *argument network*. A node can also have a number of internal attributes, denoting things such as author, textual details, certainty degree, acceptability status, etc.

Figure 1 depicts the original AIF ontology reported by Chesñevar et al [4]. The ontology has two disjoint types of nodes: *information nodes* (or I-Nodes) and *scheme nodes* (or S-Nodes). Information nodes are used to represent *passive* information contained in an argument, such as a claim, premise, data, etc. On the other hand, S-nodes capture the application of *schemes* (i.e. patterns of reasoning). Such schemes may be domain-independent patterns of reasoning, which resemble rules of inference in deductive logics but broadened to include non-deductive inference. The schemes themselves belong to a class of schemes and can be classified further into: *rule of inference scheme*, *conflict scheme*, and *preference scheme*, etc.

The AIF classifies S-Nodes further into three (disjoint) types of scheme nodes, namely *rule of inference application nodes* (RA-Node), *preference application nodes* (PA-Node) and *conflict application nodes* (CA-Node). The word ‘application’ on each of these types was introduced in the AIF as a reminder that these nodes function as in-

---

<sup>6</sup>To our knowledge, the only other OWL specification was by Bart Verheij [6] and predates the AIF.

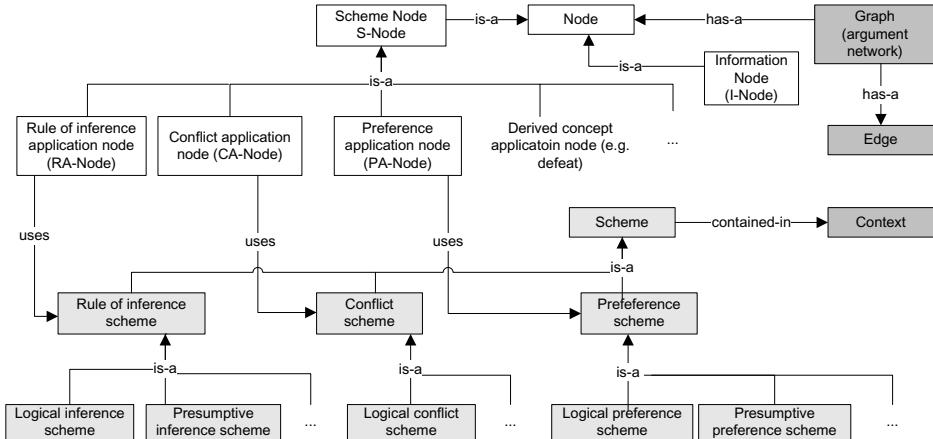


Figure 1. Original AIF Ontology [4]

stances, not classes, of possibly generic inference rules. Intuitively, RA-Nodes capture nodes that represent (possibly non-deductive) rules of inference, CA-Nodes capture applications of criteria (declarative specifications) defining conflict (e.g. among a proposition and its negation, etc.), and PA-Nodes are applications of (possibly abstract) criteria of preference among evaluated nodes. A property named “uses” expresses the fact that an instance of a scheme node *uses* a particular scheme.

The AIF core specification does not type its edges. Edge semantics can be inferred from the types of nodes they connect. The informal semantics of edges are listed in Table 1. One of the restrictions imposed by the AIF is that no outgoing edge from an I-node can be directed directly to another I-node. This ensures that the relationship between two pieces of information must be specified explicitly via an intermediate S-node.

	to <i>I-Node</i>	to <i>RA-Node</i>	to <i>PA-Node</i>	to <i>CA-Node</i>
from <i>I-Node</i>		I-node data used in applying an inference	I-node data used in applying a preference	I-node data in conflict with information in node supported by CA-node
from <i>RA-Node</i>	inferring a conclusion (claim)	inferring a conclusion in the form of an inference application	inferring a conclusion in the form of a preference application	inferring a conclusion in the form of a conflict definition application
from <i>PA-Node</i>	preference over data in I-node	preference over inference application in RA-node	meta-preferences: applying a preference over preference application in supported PA-node	preference application in supporting PA-node in conflict with preference application in PA-node supported by CA-node
from <i>CA-Node</i>	incoming conflict to data in I-node	applying conflict definition to inference application in RA-node	applying conflict definition to preference application in PA-node	showing a conflict holds between a conflict definition and some other piece of information

Table 1. Informal semantics of untyped edges in core AIF

A simple propositional logic argument network is depicted in Figure 2(a). We distinguish S-nodes from I-nodes graphically by drawing the former with a slightly thicker border. The node marked  $MP_1$  denotes an application of the modus ponens inference rule. An attack or conflict from one information or scheme node to another is captured through a CA-node, which captures the type of conflict. Since edges are directed, symmetric attack would require two sets of edges, one in each direction. Figure 2(b) depicts a symmetric conflict (through propositional negation) between two simple arguments.

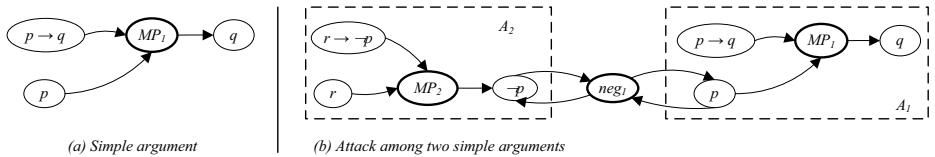


Figure 2. Examples of simple arguments

### 3. Re-Examining Scheme Reification

#### 3.1. Overview of Argument Schemes

Recently, there has been increasing interest in classifying arguments into different types (or *schemes*) based on the stereotypical inference patterns they instantiate. Many such schemes are referred to as *presumptive* inference patterns, in the sense that if the premises are true, then the conclusion may *presumably* be taken to be true. Structures and taxonomies of schemes have been proposed by many theorists (e.g. Katzav and Reed [8]). But it is Walton's exposition (e.g. recently [9]) that has been most influential in computational work. Each *Walton scheme* has a name, conclusion, set of premises and a set of critical questions. Critical questions enable contenders to identify the weaknesses of an argument based on this scheme, and potentially attack the argument. Here is an example.

#### Example 1. (*Scheme for Argument from Position to Know*)

- Assertion Premise: *E asserts that A is true (false)*
- Position to know premise: *E is in a position to know whether A is true or false;*
- Conclusion: *A may plausibly be taken to be true (false)*

Other schemes include *argument from negative consequence*, and *argument from analogy*, etc. Actual arguments are *instances* of schemes.

#### Example 2. (*Instance of Argument from Position to Know*)

- Premise: *The CIA says that Iraq has weapons of mass destruction (WMD).*
- Premise: *The CIA is in a position to know whether there are WMDs in Iraq.*
- Conclusion: *Iraq has WMDs.*

Note that premises may not always be stated, in which case we say that a given premise is *implicit* [9]. One of the benefits of argument classification is that it enables analysts to uncover the hidden premises behind an argument, once the scheme has been identified.

One way to evaluate arguments is through *critical questions*, which serve to inspect arguments based on a particular argument scheme. For example, Walton [9] identified the following critical question for “argument from position to know” (in addition to questioning the, possibly hidden, premises themselves):

#### Example 3. (*Critical Questions for Argument from Position to Know*)

1. Trustworthiness: *Is E an honest (trustworthy, reliable) source?*

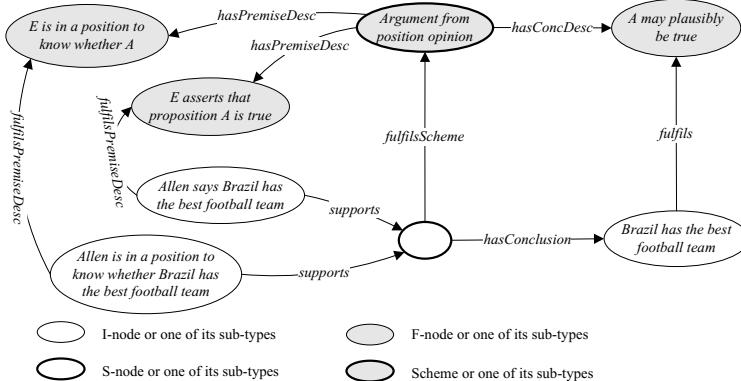
As discussed by Gordon et al [10], critical questions are not all alike. Some questions may refer to *presumptions* required for the inference to go through, while others may

refer to *exceptions* to the rule, and correspond to Toulmin's *rebuttal* [11]. The contemporary view is that the main difference between presumptions and exceptions lies in the *burden of proof*, but this is beyond the scope of the present paper.

### 3.2. Schemes in the Original AIF

The initial AIF specification separates the classification of nodes from the classification of schemes (see Figure 1). S-nodes are classified into nodes that capture inference, conflict, etc. Likewise, schemes are classified into similar sub-schemes such as inference schemes, conflict schemes, etc. S-nodes are linked to schemes via a special edge "uses."

It should be noted that the original AIF represents an "abstract model," allowing a number of different concrete reifications to be made. The reification of the AIF in the ArgDF ontology defines two classes for representing schemes and nodes [3]. Moreover, ArgDF introduced a new class, *Form node (F-node)*, to capture the generic form of statements (e.g. presumptions, premises) that constitute presumptive arguments (e.g., *PremiseDescriptor* is a sub-class of F-node that captures the generic form of premises).



**Figure 3.** An argument network linking instances of argument and scheme components

In the ArgDF ontology, actual arguments are created by instantiating nodes, while actual schemes are created by instantiating the "scheme" class. Then, argument instances (and their constituent parts) are linked to scheme instances (and their part descriptors) in order to show what scheme the argument follows. Figure 3 shows an argument network for an "argument from position to know" using the ontology of ArgDF. Each node in the argument (unshaded nodes) is explicitly linked, via a special-purpose property, to the form node it instantiates (shaded nodes). These properties (e.g. *fulfilsScheme*) are reifications of the "uses" relation between S-nodes and schemes in the AIF specification.

It is clear that ArgDF's reification of the AIF causes some redundancy. Both arguments and schemes are described with explicit structure at the instance level. Thus, the property "*fulfilsScheme*" does not capture the fact that an S-node represents an *instantiation* of some generic *class of arguments* (i.e. scheme). Having such relationship expressed explicitly can enable reasoning about the classification of schemes (as we shall demonstrate below). The ontology presented in this paper captures this relationship explicitly; presenting a simpler and more natural ontology of arguments. The AIF model is reified by interpreting schemes as classes and S-nodes as instances of those classes; in this case, the semantics of the "uses" edge can be interpreted as "*instance – of*".

### 3.3. Classification of Schemes

A notable aspect of schemes, receiving little attention in the literature, is that they do not merely describe a *flat* ontology of arguments. Consider the following scheme.

#### **Example 4. (Scheme for Appeal to Expert Opinion)**

- Expertise premise: *Source E is an expert in domain D containing proposition A.*
- Assertion premise: *E asserts that proposition A is true (false).*
- Conclusion: *A may plausibly be taken to be true (false).*

It is clear that this scheme *specialises* the scheme for argument from position to know. Apart from the fact that both schemes share the conclusion and the assertion premise, the statement “*Source E is an expert in domain D*” can be seen as a specialisation of the statement that “*E is in a position to know (things about A)*. ” Having expertise in a field causes one to be in a position to know things in that field.<sup>7</sup>

Consider also the critical questions associated with the scheme for appeal to expert opinion [9] (again, here we omit Walton’s “field” and “opinion” question since it merely questions one of the explicit premises). Notice that the trustworthiness question is repeated, while additional expertise-related questions are added.

#### **Example 5. (Critical Questions for Appeal to Expert Opinion)**

1. Expertise: *How credible is expert E?*
2. Trustworthiness: *Is E reliable?*
3. Consistency: *Is A consistent with the testimony of other experts?*
4. Backup Evidence: *Is A supported by evidence?*

Thus, schemes themselves have a hierarchical ontological structure, based on a classification of their constituent premises and conclusions. The initial AIF does not classify schemes according to this level of detail, but rather as whole entities.

## 4. A New Argument Ontology in Description Logic

Our formalisation is done using the Web ontology language OWL [5] in Description Logic (DL) notation [2] (see appendix for a short overview of DL). We use a particular dialect of OWL, called OWL DL, which is equivalent to logic  $\mathcal{SOTN}(D)$  [2].

At the highest level, we distinguish between three concepts: *statements* that can be made, *schemes* that represent classes of arguments made up of statements,<sup>8</sup> and *authors* of those statements and arguments. All these concepts are disjoint.

$$\begin{array}{lll} \text{Scheme} \sqsubseteq \text{Thing} & \text{Author} \sqsubseteq \text{Thing} & \text{Author} \sqsubseteq \neg \text{Statement} \\ \text{Statement} \sqsubseteq \text{Thing} & \text{Statement} \sqsubseteq \neg \text{Scheme} & \text{Author} \sqsubseteq \neg \text{Statement} \end{array}$$

As with the original AIF, we distinguish between rule schemes (which describe the class of arguments), conflict schemes, preference schemes, etc.

---

<sup>7</sup>Indeed, there may be other reasons to be in a position to know *A*. For example, if *E* is taken to refer to society as a whole, then the argument from position to know becomes “argument from popular opinion.”

<sup>8</sup>We use the terms “scheme” and “class of arguments” interchangeably.

*RuleScheme*  $\sqsubseteq$  *Scheme*  
*ConflictScheme*  $\sqsubseteq$  *Scheme*  
*PreferenceScheme*  $\sqsubseteq$  *Scheme*

Each of these schemes can be further classified. For example, a rule scheme may be further specialised to capture such deductive or presumptive arguments. The same can be done with different types of conflicts, preferences, and so on.

<i>DeductiveArgument</i> $\sqsubseteq$ <i>RuleScheme</i>	<i>LogicalConflict</i> $\sqsubseteq$ <i>ConflictScheme</i>
<i>PresumptiveArgument</i> $\sqsubseteq$ <i>RuleScheme</i>	<i>PresumptivePreference</i> $\sqsubseteq$ <i>PreferenceScheme</i>
<i>InductiveArgument</i> $\sqsubseteq$ <i>RuleScheme</i>	<i>LogicalPreference</i> $\sqsubseteq$ <i>PreferenceScheme</i>

We define a number of properties (or *roles* in DL terminology), which can be used to refer to additional information about instances of the ontology, such as authors of arguments, the creation date of a scheme, and so on. The domains and ranges of these properties are restricted appropriately and described below.<sup>9</sup>

<i>Scheme</i> $\sqsubseteq \forall \text{hasAuthor}.\text{Author}$	$T \sqsubseteq \forall \text{argTitle}.\text{String}$
<i>Scheme</i> $\sqsubseteq \exists \text{creationDate}$	$T \sqsubseteq \forall \text{argTitle}^-.\text{RuleScheme}$
<i>RuleScheme</i> $\sqsubseteq \exists \text{argTitle}$	$T \sqsubseteq \forall \text{authorName}.\text{String}$
$T \sqsubseteq \forall \text{creationDate}.\text{Date}$	$T \sqsubseteq \forall \text{authorName}^-.\text{Author}$
$T \sqsubseteq \forall \text{creationDate}^-.\text{Scheme}$	

To capture the structural relationships between different schemes, we first need to classify their components. We do this by classifying their premises, conclusions, presumptions and exceptions into different *classes of statements*. For example, at the highest level, we may classify statements to declarative, comparative, and imperative, etc.<sup>10</sup>

*DeclarativeStatement*  $\sqsubseteq$  *Statement*  
*ImperativeStatement*  $\sqsubseteq$  *Statement*  
*ComparativeStatement*  $\sqsubseteq$  *Statement* ...

Actual statement instances have a property that describes their textual content.

$T \sqsubseteq \forall \text{claimText}.\text{String}$   
 $T \sqsubseteq \forall \text{claimText}^-.\text{Statement}$

When defining a particular RuleScheme (i.e. class of arguments), we capture the relationship between each scheme and its components. Each argument has exactly one conclusion and at least one premise (which are, themselves, instances of class “Statement”). Furthermore, presumptive arguments may have presumptions and exceptions.

*RuleScheme*  $\sqsubseteq \forall \text{hasConclusion}.\text{Statement}$   
*RuleScheme*  $\sqsubseteq \exists \text{hasConclusion}$   
*RuleScheme*  $\sqsubseteq \forall \text{hasPremise}.\text{Statement}$   
*RuleScheme*  $\sqsupseteq \exists \text{hasPremise}$

---

<sup>9</sup>The range and domain of property *R* are described using  $T \sqsubseteq \forall R.C$  and  $T \sqsubseteq \forall R^- . C$  (see appendix).

<sup>10</sup>We avoid an ontological discussion of all types of statements. Our interest is in a (humble) demonstration of how a classification of argument *parts* may help automate reasoning about argument *types*. How individual parts get categorised into classes (e.g. using automated or manual tagging) is beyond the scope of this paper.

*PresumptiveArgument*  $\sqsubseteq \forall \text{hasPresumption.Statement}$

*PresumptiveArgument*  $\sqsubseteq \forall \text{hasException.Statement}$

With this in place, we can further classify the above statement types to cater for a variety of schemes. For example, to capture the scheme for “argument from position to know,” we first need to define the following classes of declarative statements. Each class is listed an OWL-DL *annotation property* called `formDescription` which describes the statement’s typical form. Annotation properties are used to add meta-data about classes.

*PositionToHaveKnowledgeStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “E is in position to know whether A is true (false)”

*KnowledgeAssertionStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “E asserts that A is true(false)”

*KnowledgePositionStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “A may plausibly be taken to be true(false)”

*LackOfReliabilityStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “E is not a reliable source”

Now we are ready to fully describe the scheme for “argument from position to know.” The following are the necessary as well as the necessary-and-sufficient conditions for an instance to be classified as an argument from position to know.

*ArgFromPositionToKnow*  $\equiv (\text{PresumptiveArgument} \sqcap$

$\exists \text{hasConclusion.KnowledgePositionStmnt} \sqcap$

$\exists \text{hasPremise.PositionToHaveKnowledgeStmnt} \sqcap$

$\exists \text{hasPremise.KnowledgeAssertionStmnt})$

*ArgFromPositionToKnow*  $\sqsubseteq \exists \text{hasException.LackOfReliabilityStmnt}$

Now, for the “appeal to expert opinion” scheme, we only need to define one additional premise type, since both the conclusion and the assertion premise are identical to those of “argument from position to know.”

*FieldExpertiseStmnt*  $\sqsubseteq \text{PositionToHaveKnowledgeStmnt}$

`formDescription` : “source E is an expert in subject domain D containing proposition A”

Similarly, one of the exceptions of this scheme is identical to “argument from position to know.” The remaining presumptions and exception are added as follows:

*ExpertiseInconsistencyStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “A is not consistent with other experts assertions”

*CredibilityOfSourceStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “E is credible as an expert source”

*ExpertiseBackUpEvidenceStmnt*  $\sqsubseteq \text{DeclarativeStatement}$

`formDescription` : “E’s assertion is based on evidence”

Likewise, the necessary-and-sufficient conditions of “appeal to expert opinion” are:

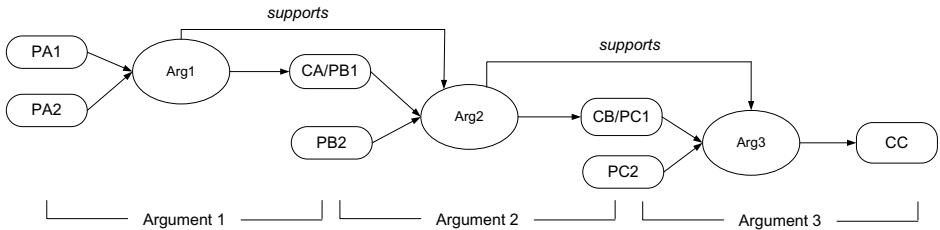
*AppToExpertOpinion*  $\equiv (\text{PresumptiveArgument} \sqcap$

$\exists \text{hasConclusion.KnowledgePositionStmnt} \sqcap$

$\exists \text{hasPremise.FieldExpertiseStmnt} \sqcap \exists \text{hasPremise.KnowledgeAssertionStmnt})$

*AppToExpertOpinion*  $\sqsubseteq \exists \text{hasException.LackOfReliabilityStmnt}$

*AppToExpertOpinion*  $\sqsubseteq \exists \text{hasException.ExpertiseInconsistencyStmnt}$



**Figure 4.** Support among chained arguments

*AppToExpertOpinion*  $\sqsubseteq \exists \text{hasPresumption}.\text{CredibilityOfSourceStmnt}$

*AppToExpertOpinion*  $\sqsubseteq \exists \text{hasPresumption}.\text{ExpertiseBackUpEvidenceStmnt}$

It is important to note that a single statement instance might adhere to different types of premises, conclusions or even presumptions and exceptions as the ontology should enable re-using existing statement instances and creating interlinked and dynamic argument networks.

## 5. OWL Reasoning over Argument Structures

In this section, we discuss ways in which the expressive power of OWL and its support for reasoning can be used to enhance user interaction with arguments. We focus on features that extend our previous work on the RDF Schema-based ArgDF system [3].

### 5.1. Inference of Indirect Support in Chained Arguments

One of the advantages of OWL over RDF Schema is that OWL supports inference over transitive properties. In other words, if  $r(X, Y)$  and  $r(Y, Z)$ , then OWL reasoners can infer  $r(X, Z)$ . This can be used to enhance argument querying.

Arguments can support other arguments by supporting their premises. This results in argument *chaining* where a claim acts both as a premise of one argument and as a conclusion of another. This situation is illustrated in Figure 4. In Argument 1, premises PA1 and PA2 have the conclusion CA which is used at the same time as premise PB1 of the argument 2. Premises PB1 and PB2 have the conclusion CB which is used at the same time as premise PC1 of argument 3; PC1 and PC2 have the conclusion CC. Here, we can say that Argument 1 *indirectly* supports Argument 3.

A user may wish to retrieve all arguments that directly or indirectly support conclusion CC. RDF Schema does not provide straightforward support for retrieving this information. We added a transitive property *supports* to the ontology, linking the supporting argument to the supported argument in a chain: *RuleScheme*  $\sqsubseteq \forall \text{supports}.\text{RuleScheme}$ . By using this edge, and the description logic reasoner, small and elegant queries can retrieve the desired information.

### 5.2. Automatic Classification of Argument Schemes and Instances

As explained above, due to the hierarchy of specialisation among different descriptors of scheme components (i.e. statements) as well as the necessary and sufficient condi-

tions defined on each scheme, it is possible to infer the classification hierarchy among schemes.

**Example 6. (Inferring scheme relationships)** Following from the statement and scheme definitions of “appeal to expert opinion” and “argument from position to know” outlined earlier, the reasoner infers that the former is a sub-class of the latter.

Similar inferences can be undertaken over other classes. A more elaborate example involves inferring the “fear appeal argument” scheme as sub-class of “argument from negative consequence.” Consider the specification of the argument schemes of “argument from negative consequence” and “fear appeal argument.” The necessary-and-sufficient part of scheme description of the above arguments are detailed as follows.

$$\begin{aligned} \text{ArgNegativeConseq} &\equiv (\text{PresumptiveArgument} \sqcap \\ &\quad \exists \text{hasConclusion}.\text{ForbiddenActionStmnt} \sqcap \exists \text{hasPremise}.\text{BadConsequenceStmnt}) \\ \text{FearAppealArg} &\equiv (\text{PresumptiveArgument} \sqcap \exists \text{hasConclusion}.\text{ForbiddenActionStmnt} \sqcap \\ &\quad \exists \text{hasPremise}.\text{FearfulSituationStmnt} \sqcap \\ &\quad \exists \text{hasPremise}.\text{FearedBadConsequenceStmnt}) \end{aligned}$$

The statements are defined as follows. Note that the “Feared Bad Consequence” statement is a specialisation of “Bad Consequence” statement, since it limits the bad consequence to those portrayed in the fearful situation.

$$\begin{aligned} \text{BadConsequenceStmnt} &\sqsubseteq \text{DeclarativeStatement} \\ \text{formDescription} &: \text{“If A is brought about, bad consequences will plausibly occur”} \\ \text{ForbiddenActionStmnt} &\sqsubseteq \text{DeclarativeStatement} \\ \text{formDescription} &: \text{“A should not be brought about”} \\ \text{FearfulSituationStmnt} &\sqsubseteq \text{DeclarativeStatement} \\ \text{formDescription} &: \text{“Here is a situation that is fearful to you”} \\ \text{FearedBadConsequenceStmnt} &\sqsubseteq \text{BadConsequenceStmnt} \\ \text{formDescription} &: \text{“If you carry out A, then the negative consequences portrayed in this fearful situation will happen to you”} \end{aligned}$$

As a result of classification of scheme hierarchies, instances belonging to a certain scheme class will also be inferred to belong to all its super-classes. For example, if the user queries to return all instances of “argument from negative consequences,” the instances of all specializations of the scheme, such as all argument instances from “fear appeal arguments” are also returned.

### 5.3. Inferring Critical Questions

Since the schemes are classified by the reasoner into a hierarchy, if certain presumptions or exceptions are not explicitly stated for a specific scheme but are defined on any of its super-classes, the system is able to infer and add those presumptions and exceptions to instances of that specific scheme class. Consider the critical questions for “fear appeal argument” and “argument from negative consequence” described below.

**Example 7. (Critical Questions for Fear Appeal Argument)**

1. *Should the situation represented really be fearful to me, or is it an irrational fear that is appealed to?*

2. If I don't carry out A, will that stop the negative consequence from happening?
3. If I do carry out A, how likely is it that the negative consequence will happen?

**Example 8. (Critical Questions for Argument From Negative Consequence)**

1. How strong is the probability or plausibility that these cited consequence will (may, might, must) occur?
2. What evidence, if any, supported the claim that these consequence will (may, might, must) occur if A is brought about?
3. Are there consequence of the opposite value that ought to be taken into account?

“Fear appeal argument” is classified as a sub-class of “argument from negative consequence.” The critical questions 2 and 3 of “argument from negative consequence” have not been explicitly defined on “fear appeal argument,” but can be inferred. Since critical questions provide a way for evaluation of an argument, inferring such additional questions for can enhance the analysis process.

## 6. Implementation

In this section, we describe our implementation (in-progress) of a Web-based system for creating, manipulating, and querying complex argument structures. The core Website is built on Java. Jena<sup>11</sup> provides the programming environment for manipulating the ontology model. Moreover, ARQ<sup>12</sup> libraries are used to provide the SPARQL[12] query engine. Pellet [13], an open source description logic reasoner for OWL-DL enables inference over the ontology model generated by Jena. The ontology and instances are stored in an SQL Server database. In brief, the new implementation offers the following features:

- Creation of new semantically annotated arguments, using new or existing authored statements. While this feature was implemented in ArgDF, the new system uses subsumption reasoning to infer and add critical questions to each new argument instance.
- Attacking and supporting parts of existing arguments.
- Retrieving supporting or attacking arguments/claims for a given claim. In case of support, both direct and indirect supporting arguments are listed.
- Retrieving scheme details, in order to inspect them, such as the conclusion, premise, presumption or exception descriptors as well as the scheme’s inferred super-class(es) and sub-class(es).
- Creation of new schemes through the user interface.
- Search for arguments based on keywords, authors, schemes. When searching for arguments of a specific scheme type, inference is used to return all the arguments that are instances of that specific scheme as well as instances that belong to any of its sub-classes.

Although some of the above features have already appeared in ArgDF, the key feature of the current implementation is its use of OWL inference to enhance the retrieval of arguments (as described in detail in Section 5).

---

<sup>11</sup><http://jena.sourceforge.net/>

<sup>12</sup><http://jena.sourceforge.net/ARQ/>

## 7. Conclusion

We reported on ongoing work that exploits the OWL language for creating, navigating and manipulating complex argument structures. The new ontology enhances our previous RDF Schema-based implementation [3]. In particular, we now model schemes as classes (as opposed to instances), which enables detailed classification of schemes themselves. Secondly, our new system enables the first explicit use of Description Logic-based OWL reasoning for classifying arguments and schemes in a Web-based system. This provides a seed for further work that combines traditional argument-based reasoning techniques [7] with ontological reasoning in a Semantic Web environment. Once the system implementation is complete, we aim to focus on content acquisition. We will explore the integration of arguments from other repositories into our system. We will also work on integrating effective argument visualisation techniques, which can help in acquisition as well as interaction with the argument repository.

## Acknowledgement

We are grateful for the detailed comments received from the anonymous reviewers.

## References

- [1] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, pages 29–37, May 2001.
- [2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, Cambridge, UK, 2003.
- [3] I. Rahwan, F. Zablith, and C. Reed. Laying the foundations for a world wide argument web. *Artificial Intelligence*, 171(10–15):897–921, 2007.
- [4] C. I. Chesñevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, and S. Willmott. Towards an argument interchange format. *The Knowledge Engineering Review*, 21(4):293–316, 2007.
- [5] D. L. McGuinness and F. van Harmelen. OWL web ontology language overview. W3C Recommendation REC-owl-features-20040210/, World Wide Web Consortium (W3C), February 2004.
- [6] B. Verheij. An argumentation core ontology as the centerpiece of a myriad of argumentation formats. Technical report, Agentlink Argumentation Interchange Format Technical Forum, 2005.
- [7] C. I. Chesñevar, A. Maguitman, and R. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.
- [8] C. Reed and J. Katzav. On argumentation schemes and the natural classification of arguments. *Argumentation*, 18(2):239–259, 2004.
- [9] D. N. Walton. *Fundamentals of Critical Argumentation*. Cambridge University Press, New York, USA, 2006.
- [10] T. F. Gordon, H. Prakken, and D. Walton. The carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10–15):875–896, 2007.
- [11] S. Toulmin. *The Uses of Argument*. Cambridge University Press, Cambridge, UK, 1958.
- [12] E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. W3C Candidate Recommendation CR-rdf-sparql-query-20070614, World Wide Web Consortium (W3C), 2007.
- [13] E. Sirin, B. Parsia, B. Cuenca Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical OWL-DL reasoner. *Web Semantics*, 5(2):51–53, 2007.

## Appendix: Description Logics

Description Logics (DLs) [2] are a family of knowledge representation languages which can be used to represent the terminological knowledge of an application domain. The idea is to define complex concept hierarchies from basic (atomic) concepts, and to define complex roles (or properties) that define relationships between concepts.

Table 2 shows the syntax and semantics of common concept and role constructors. The letters  $A, B$  are used for atomic concepts and  $C, D$  for concept descriptions. For roles, the letters  $R$  and  $S$  are used and non-negative integers (in number restrictions) are denoted by  $n, m$  and individuals (i.e. instances) by  $a, b$ . An interpretation  $\mathcal{I}$  consists of a non-empty set  $\Delta^{\mathcal{I}}$  (the domain of the interpretation) and an interpretation function, which assigns to every atomic concept  $A$  a set  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$  and to every atomic role  $R$  a binary relation  $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ .

A DL knowledge base consists of a set of terminological axioms (often called *TBox*) and a set of assertional axioms or assertions (often called *ABox*). A finite set of definitions is called a *terminology* or *TBox* if the definitions are unambiguous, i.e., no atomic concept occurs more than once as left hand side.

Name	Syntax	Semantics
<b>Concept &amp; Role Constructors</b>		
Top	$\top$	$\Delta^{\mathcal{I}}$
Bottom	$\perp$	$\emptyset$
Concept Intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Concept Union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Concept Negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
Value Restriction	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a, b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$
Existential Quantifier	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.(a, b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$
Unqualified	$\geq nR$	$\{a \in \Delta^{\mathcal{I}} \mid   \{b \in \Delta^{\mathcal{I}} \mid (a, b) \in R^{\mathcal{I}}\}   \geq n\}$
Number	$\leq nR$	$\{a \in \Delta^{\mathcal{I}} \mid   \{b \in \Delta^{\mathcal{I}} \mid (a, b) \in R^{\mathcal{I}}\}   \leq n\}$
Restriction	$= nR$	$\{a \in \Delta^{\mathcal{I}} \mid   \{b \in \Delta^{\mathcal{I}} \mid (a, b) \in R^{\mathcal{I}}\}   = n\}$
Role-value-map	$R \subseteq S$ $R = S$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a, b) \in R^{\mathcal{I}} \rightarrow (a, b) \in S^{\mathcal{I}}\}$ $\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a, b) \in R^{\mathcal{I}} \leftrightarrow (a, b) \in S^{\mathcal{I}}\}$
Nominal	$I$	$I^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ with $  I^{\mathcal{I}}   = 1$
Universal Role	$U$	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
Role Intersection	$R \sqcap S$	$R^{\mathcal{I}} \cap S^{\mathcal{I}}$
Role Union	$R \sqcup S$	$R^{\mathcal{I}} \cup S^{\mathcal{I}}$
Role Complement	$\neg R$	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}$
Role Inverse	$R^{-}$	$\{(b, a) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a, b) \in R^{\mathcal{I}}\}$
Transitive Closure	$R^{+}$	$\bigcup_{n>1} (R^{\mathcal{I}})^n$
Role Restriction	$R c$	$R^{\mathcal{I}} \cap (\Delta^{\mathcal{I}} \times C^{\mathcal{I}})$
Identity	$id(C)$	$\{(d, d) \mid d \in C^{\mathcal{I}}\}$
<b>Terminological Axioms</b>		
Concept Inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
Concept Equality	$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$
Role Inclusion	$R \sqsubseteq S$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$
Role Equality	$R \equiv S$	$R^{\mathcal{I}} = S^{\mathcal{I}}$

Table 2. Some Description Logic Role Constructors, Concept Constructors, and Terminological Aximos

To give examples of what can be expressed in DLs, we suppose that *Person* and *Female* are atomic concepts. Then  $\text{Person} \sqcap \text{Female}$  is DL concept describing, intuitively, those persons that are female. If, in addition, we suppose that *hasChild* is an atomic role, we can form the concept  $\text{Person} \sqcap \exists \text{hasChild}$ , denoting those persons that have a child. Using the bottom concept, we can also describe those persons without a child by the concept  $\text{Person} \sqcap \forall \text{hasChild}.\perp$ . These examples show how we can form complex descriptions of concepts to describe classes of objects.

The *terminological axioms* make statements about how concepts or roles are related to each other. It is possible to single out definitions as specific axioms and identify terminologies as sets of definitions by which we can introduce atomic concepts as abbreviations or names for complex concepts.

An equality whose left-hand side is an atomic concept is a *definition*. Definitions are used to introduce symbolic names for complex descriptions. For instance, by the axiom  $\text{Mother} \equiv \text{Woman} \sqcap \exists \text{hasChild}.\text{Person}$ , we associate to the description on the right-hand side the name *Mother*. Symbolic names may be used as abbreviations in other descriptions. If, for example, we have defined *Father* analogously to *Mother*, we can define *Parent* as  $\text{Parent} \equiv \text{Mother} \sqcup \text{Father}$ . Table 3 shows a terminology with concepts concerned with family relationships.

The sentence  $\top \sqsubseteq \forall \text{hasParent}.\text{Person}$  expresses that the range of the property *hasParent* is the class *Person* (more technically, if the property *hasParent* holds between any concept and another concept, the latter concept must be of type *Person*).

Name	DL Syntax	Example
<b>Constructor / axiom</b>		
Concept Intersection	$C \sqcap D$	$\text{Woman} \equiv \text{Person} \sqcap \text{Female}$
Concept Union	$C \sqcup D$	$\text{Parent} \equiv \text{Mother} \sqcup \text{Father}$
Concept Negation	$\neg C$	$\text{Man} \equiv \text{Person} \sqcap \neg \text{Woman}$
Existential Quantifier	$\exists R.C$	$\text{Mother} \equiv \text{Woman} \sqcap \exists \text{hasChild}.\text{Person}$
Value Restriction	$\forall R.C$	$\text{MotherWithoutSons} \equiv \text{Mother} \sqcap \forall \text{hasChild}.\neg \text{Woman}$
MinCardinality	$\geq nR$	$\text{MotherWithAtLeastThreeChildren} \equiv \text{Mother} \sqcap \geq 3 \text{hasChild}$
Cardinality	$= nR$	$\text{FatherWithOneChild} \equiv \text{Father} \sqcap = 1 \text{hasChild}$
Bottom	$\perp$	$\text{PersonWithoutAChild} \equiv \text{Person} \sqcap \forall \text{hasChild}.\perp$
Transitive Property	$R^+ \sqsubseteq R$	$\text{ancestor}^+ \sqsubseteq \text{ancestor}$
Role Inverse	$R \equiv S^-$	$\text{hasChild} \equiv \text{hasParent}^-$
Concept Inclusion	$C \sqsubseteq D$	$\text{Woman} \sqsubseteq \text{Person}$
Disjoint with	$C \sqsubseteq \neg D$	$\text{Man} \sqsubseteq \neg \text{Woman}$
Role Inclusion	$R \sqsubseteq S$	$\text{hasDaughter} \sqsubseteq \text{hasParent}$
Range	$\top \sqsubseteq \forall R.C$	$\top \sqsubseteq \forall \text{hasParent}.\text{Person}$
Domain	$\top \sqsubseteq \forall R^-C$	$\top \sqsubseteq \forall \text{hasParent}^-.\text{Person}$

**Table 3.** A terminology (TBox) with concepts about family relationships

# AIF<sup>+</sup>: Dialogue in the Argument Interchange Format

Chris Reed, Simon Wells, Joseph Devereux & Glenn Rowe

*School of Computing, University of Dundee, Dundee DD1 4HN, UK*

**Abstract.** This paper extends the Argument Interchange Format to enable it to represent dialogic argumentation. One of the challenges is to tie together the rules expressed in dialogue protocols with the inferential relations between premises and conclusions. The extensions are founded upon two important analogies which minimise the extra ontological machinery required. First, locutions in a dialogue are analogous to AIF I-nodes which capture propositional data. Second, steps between locutions are analogous to AIF S-nodes which capture inferential movement. This paper shows how these two analogies combine to allow both dialogue protocols and dialogue histories to be represented alongside monologic arguments in a single coherent system.

**Keywords.** Argumentation, Dialogue, Interchange, Standards

## 1. Introduction and Background

Research into argumentation in AI has enjoyed rapid growth over the past ten years, made all the more distinctive by the fact that argumentation-based models have had impact right across the board, from natural language processing, through knowledge representation and multi-agent communication, to automated reasoning and computer support collaborative working. One of the challenges of such wide-ranging models is the need for interoperability – to ensure that the resources developed through argumentation-based CSCW can utilise advances in argumentation KR; to ensure that new models of argument representation can yield to argument-based reasoning mechanisms; to ensure that autonomously created arguments can be communicated using argument-based protocols.

The Argument Interchange Format (AIF) [1] was developed to tackle this challenge. The aim was to develop a means of expressing argument that would provide a flexible – yet semantically rich – way of representing argumentation structures. The AIF was put together to try to harmonise the strong formal tradition initiated to a large degree by [2], the natural language research described at CMNA workshops since 2000, and the multi-agent argumentation work that has emerged from the philosophy of [18], amongst others.

The AIF to date has had significant impact in practical development projects such as the WWAW [9], but in its 2006 form had not integrated modelling of dialogue. This has meant that one important strand of research (spanning phi-

losophy, linguistics, natural language engineering and multi-agent systems) has been excluded from the benefits that the AIF affords. Modgil and McGinnis [6] go some way to addressing this omission. The current paper builds on their work integrating other, more linguistically oriented, models (such as [10] and [20]) to show how an extended version of the AIF, which we call AIF<sup>+</sup>, can handle dialogic argumentation in the same, broad way that the AIF simpliciter currently handles monologic argument. The motivation for this work can be summarised through philosopher Daniel J. O'Keefe's distinction between argument\_1 and argument\_2: argument\_1 is an arrangement of claims into a monological structure, whilst argument\_2 is a dialogue between participants - as O'Keefe [7][p122] puts it, "The distinction here is evidenced in everyday talk by ... the difference between the sentences 'I was arguing\_1 that P' and 'we were arguing\_2 about Q.' " Clearly there are links between argument\_1 and argument\_2 in that the steps and moves in the latter are constrained by the dynamic, distributed and inter-connected availability of the former, and further in that valid or acceptable instances of the former can come about through sets of the latter. An understanding of these intricate links which result from protocols and argument-based knowledge demands a representation that handles both argument\_1 and argument\_2 coherently. It is this that the AIF<sup>+</sup> sets out to provide. There are several specific goals for this work:

1. To extend the AIF so that it can support representation of argumentation protocols (i.e. specifications of how dialogues are to proceed).
2. To extend the AIF so that it can support representation of dialogue histories (i.e. records of how given dialogues did proceed).
3. To place little or no restriction on the types of dialogue protocol and dialogue history that can be represented.
4. To integrate the dialogic argument representation of the AIF<sup>+</sup> with the monologic argument representation of the AIF.
5. To meet all of 1-4 with the minimum extra representational machinery possible.

There are thus many interesting issues that are *not* being tackled here. Like the AIF, the AIF<sup>+</sup> is concerned with representation, not processing. So although there may be any number of software systems that allow the creation or execution of dialogues that conform to a protocol, or that allow the determination of whether or not a given dialogue conforms to a dialogue protocol, the business of the AIF<sup>+</sup> is just to represent the data. (This is analogous to the path taken in the AIF: though, for example, it may be possible for a system to compute acceptability according to a given semantics, the task of the AIF is simply to represent the arguments in such a way that acceptability computations can be performed easily). Similarly, the AIF<sup>+</sup> maintains a clear separation between the representation of prescriptive or normative structures (in protocol specification) and the representation of actual arguments (in dialogue histories). This again follows the successful pattern of the AIF, wherein the normative structures of inference (in inference rules, schemes, patterns of conflict and preference and so on), are separate from the characterisation of how individual arguments do in fact stand in relation to one another. Finally, though the AIF<sup>+</sup> is interested in repre-

senting how dialogues should proceed and how they have proceeded, there is no representation of the processing required by, for example, an agent that allowed it to decide what should be said. Again, such tactical and strategic processing is beyond the scope of what the AIF<sup>+</sup> should handle.

## 2. The AIF

The AIF can be seen as a representation scheme constructed in three layers. At the most abstract layer, the AIF provides an ontology of concepts which can be used to talk about argument structure. This ontology describes an argument by conceiving of it as a network of connected nodes that are of two types: information nodes that capture data (such as datum and claim nodes in a Toulmin analysis, or premises and conclusions in a traditional analysis), and scheme nodes that describe passage between information nodes (similar to warrants or rules of inference). Scheme nodes in turn come in several different guises, including scheme nodes that correspond to support or inference (or rule application nodes), scheme nodes that correspond to conflict or refutation (or conflict application nodes), and scheme nodes that correspond to value judgements or preference orderings (or preference application nodes). At this topmost layer, there are various constraints on how components interact: information nodes, for example, can only be connected to other information nodes via scheme nodes of one sort or another. Scheme nodes, on the other hand, can be connected to other scheme nodes directly (in cases, for example, of arguments that have inferential components as conclusions, e.g. in patterns such as Kienpointner's [3] "warrant-establishing arguments"). The AIF also provides, in the extensions developed for the WAWA [9], the concept of a "Form" (as distinct from the "Content" I- and S-nodes). Forms allow the ontology to represent uninstantiated definitions of schemes at the next layer down.

A second, intermediate layer provides a set of specific argumentation schemes (and value hierarchies, and conflict patterns). Thus, the uppermost layer in the AIF ontology lays out that presumptive argumentation schemes are types of rule application nodes, but it is the intermediate layer that cashes those presumptive argumentation schemes out into Argument from Consequences, Argument from Cause to Effect and so on. At this layer, the form of specific argumentation schemes is defined: each will have a conclusion description (such as "A may plausibly be taken to be true") and one or more premise descriptions (such as "E is an expert in domain D").

It is also at this layer that, as [9] have shown, the AIF supports a sophisticated representation of schemes and their critical questions. In addition to descriptions of premises and conclusions, each presumptive inference scheme also specifies descriptions of its presumptions and exceptions. Presumptions are represented explicitly as information nodes, but, as some schemes have premise descriptions that entail certain presumptions, the scheme definitions also support entailment relations between premises and presumptions.

Finally the third and most concrete level supports the integration of actual fragments of argument, with individual argument components (such as strings

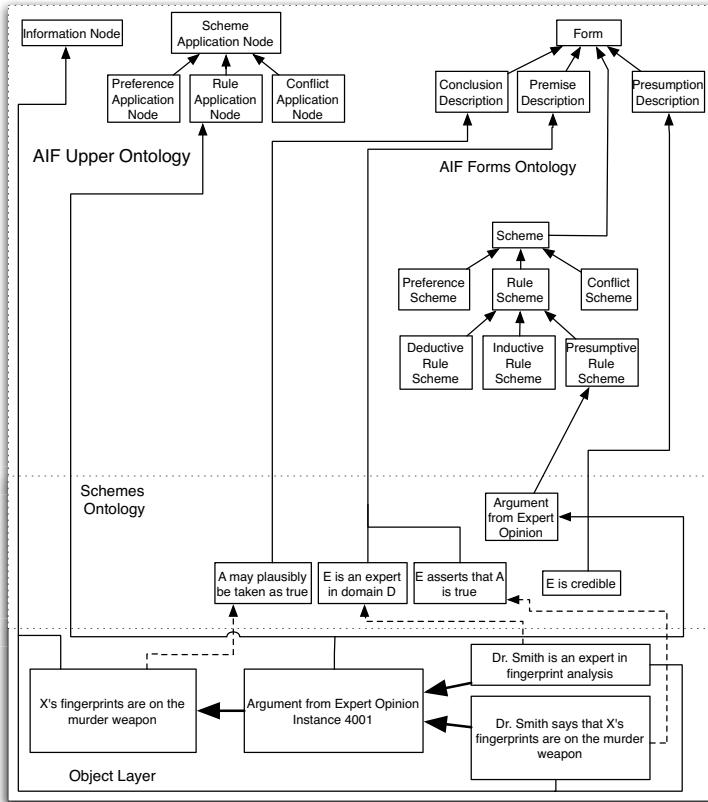
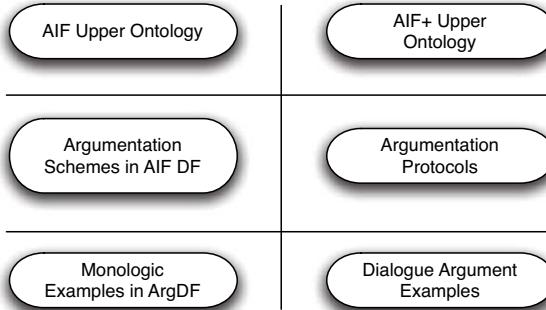


Figure 1. Three levels of AIF representation.

of text) instantiating elements of the layer above. At this third layer an actual instance of a given scheme is represented as a rule application node – the terminology now becomes clearer. This rule application node is said to fulfill one of the presumptive argumentation scheme descriptors at the level above. As a result of this fulfillment relation, premises of the rule application node fulfill the premise descriptors, the conclusion fulfills the conclusion descriptor, presumptions can fulfill presumption descriptors, and conflicts can be instantiated via instances of conflict schemes that fulfill the conflict scheme descriptors at the level above. Again, all the constraints at the intermediate layer are inherited, and new constraints are introduced by virtue of the structure of the argument at hand. Figure 1 shows diagrammatically how all of these pieces fit together (but it should be borne in mind that figure 1 aims to diagram how the AIF works - it is a poor diagram of the argument that is represented). The “fulfills” relationships are indicated by dotted lines, and inference between object layer components (i.e. the arguments themselves) by thick lines. Remaining lines show “is-a” relationships. Note that many details have been omitted for clarity, including the way in which scheme descriptions are constructed from forms.



**Figure 2.** Ontological Architecture

### 3. Architecture

The aim for the AIF<sup>+</sup> is to preserve and exploit the three levels of representation developed in the AIF. First, the upper ontology will need extending to cope with concepts that are unique to dialogue. Second, we will need some examples of dialogue protocol forms to govern what happens at the object level. And finally, we will need an example that provides the argument material conforming to the protocol and using the concepts from the upper ontology. This mapping is summarised in figure 2 and forms the structure for the remainder of the paper.

### 4. AIF<sup>+</sup>: Ontological Extensions

We base the construction of the ontological extensions required for the AIF<sup>+</sup> on the expanded version of the AIF presented in [9], and specifically, upon the treatment there of argumentation schemes.

The fundamental building blocks of dialogues are the individual locutions. In the context of the AIF, Modgil and McGinnis [6] have proposed modelling locutions as I-nodes. We follow this approach primarily because statements about locution events are propositions that could be used in arguments. So for example, the proposition, *'Joseph says, 'COMMA-08 will be in Toulouse'* could be referring to something that happened in a dialogue (and later we shall see how we might therefore wish to reason about the proposition, *'COMMA-08 will be in Toulouse'*) – but it might also play a role in another, monologic argument (say, an argument from expert opinion, or just an argument about Joseph's communicative abilities).

Associating locutions exactly with I-nodes, however, is insufficient. There are several features that are unique to locutions, and that do not make sense for propositional information in general. Foremost amongst these features is that locutions often have propositional content (there are, arguably, exceptions, such as the locutions, 'Yes,' and 'No'). The relationship between a locution and the proposition it employs is, as Searle [14] argues, constant - i.e. "propositional content" is a property of (some) locutions. Whilst other propositions, such as might be expressed in other I-nodes, may also relate to further propositions, (e.g. the

proposition, *It might be the case that it will rain*) there is no such constant relationship of propositional content. On these grounds, we should allow representation of locutions to have propositional content, but not allow it for I-nodes in general – and therefore the representation of locutions should form a subclass of I-nodes in general. We call this subclass L-nodes. There are further reasons for distinguishing L-nodes as a special case of I-nodes, such as the identification of which dialogue(s) a locution is part of. (There are also some features which one might expect to be unique to locutions, but on reflection are features of I-nodes in general. Consider, for example, a time index - we may wish to note that Joseph said, ‘COMMA-08 will be in Toulouse’ at 10am exactly on the 1st October 2007. Such specification, however, is common to all propositions. Strictly speaking, *It might be the case that it will rain* is only a proposition if we spell out where and when it holds. In other words, a time index could be a property of I-nodes in general, though it might be rarely used for I-nodes and often used in L-nodes).

Given that locutions are a (subclass of) I-nodes, according to the AIF<sup>+</sup>, they can only be connected through S-nodes. Modgil and McGinnis [6] do not tackle ontological concerns directly, but assume that a new type of S-node will suffice. There is a missed opportunity here. There is a direct analogy between the way in which two I-nodes are inferentially related when linked by an RA-node, and the way in which two L-nodes are related when one responds to another by the rules of a dialogue. Imagine, for example, a dialogue in which Joseph says, ‘COMMA-08 will be in Toulouse’ and Simon responds by asking, ‘Why is that so?’. In trying to understand what has happened, one could ask, ‘Why did Simon ask his question?’ Now although there may be many motivational or intentional aspects to an answer to this question, there is at least one answer we could give purely as a result of the dialogue protocol, namely, ‘Because Joseph had made a statement’. That is to say, there is plausibly an inferential relationship between the proposition, ‘Joseph says COMMA-08 will be in Toulouse’ and the proposition, ‘Simon asks why it is that COMMA-08 will be in Toulouse’. That inferential relationship is similar to a conventional inferential relationship, as captured by an RA node. Clearly, though, the grounds of such inference lie not in a scheme definition, but in the protocol definition. Specifically, the inference between two L-nodes is governed by a *transition*, so a given inference is a specific application of a transition. Hence we call such nodes transition application nodes (TA-nodes), and define TA-nodes as a subclass of RA-nodes.

Finally, by analogy to the ontological machinery of argumentation schemes, we can view transitions as forms that are fulfilled by TA nodes. These transitions, however are not all there is to a protocol. A protocol defined by a set of transitions in this way is equivalent in power to a finite state automaton (though the transitions in AIF<sup>+</sup> correspond to transition-state-transition tuples in an FSA). Alternative models of protocol composition (such as the declarative language LCC [12], or the representation techniques of, e.g. Dooley graphs [8], or the use of commitment-based semantics [18] and their computational representation [19], [20]) range in sophistication from finite state to Turing complete. In order to represent these protocols in full, therefore, more is required. The most straightforward approach derives from AI planning, specifying pre- and post-conditions on operators that correspond to locutions. These protocol components specify the

general forms that locutions can take, and are composed to form transition forms. But in addition, many protocols associate additional constraints with what are here called transitions. A good example is Mackenzie's DC protocol [4], which constrains, for example the *Resolve* locution, when coming after a *Why* locution such that the content of the latter must be an “immediate consequence” of the former. Immediate consequence is a logical notion, but one which only comes into play in the response by one particular locution to another. This specific transition scheme can thus be interpreted as having a presumption (about immediate consequence) in much the same way that specific inference schemes have presumptions (about, for example, the veracity of an expert).

So, in just the same way that an RA-node fulfils a rule of inference scheme form, and the premises of that RA-node fulfil the premise descriptions of the scheme form, so too, a TA-node fulfils a transitional inference scheme form, and the locutions connected by that TA-node fulfil the locution descriptions of the scheme form. The result is that all of the machinery for connecting the normative, prescriptive definitions in schemes with the actual, descriptive material of a monologic argument is re-used to connect the normative, prescriptive definitions of protocols with the actual, descriptive material of a dialogic argument.

With these introductions, the upper ontology for AIF<sup>+</sup> is complete. For both I-nodes and RA-nodes, we need to distinguish between the old AIF class and the new subclass which contains all the old I-nodes and RA-nodes excluding L-nodes and TA-nodes (respectively). As the various strands and implementations of AIF continue, we will want to continue talking about I-nodes and RA-nodes and in almost all cases, it is the superclass that will be intended. We therefore keep the original names for the superclasses (I-node and RA-node), and introduce the new subclasses I' and RA' for the sets (I-nodes L-nodes) and (RA-nodes TA-nodes) respectively. The upper ontology is thus as in figure 3.

## 5. Protocol Representation

To show how the AIF<sup>+</sup> supports both argument\_1 and argument\_2 in such a way that the links between them can be captured, we need an example of a dialogue protocol. For this initial exploration, we need a protocol that is sufficiently simple to be clear, whilst sufficiently sophisticated to exercise the capabilities of the AIF<sup>+</sup>. A suitable protocol can be found in [11] which extends a simple dialectical game based upon the formal game CB [17] to incorporate argumentation schemes and critical questions. This protocol is called Argumentation Scheme Dialogue, or ASD – but it is important to emphasise, that this is simply an example of a protocol that can be represented in AIF<sup>+</sup>. We are not arguing either for the utility of ASD, nor for any special role for it in the general theory of AIF<sup>+</sup>. The rules of ASD are as follows:

### Locution Rules

- i. **Statements** Statement letters, S, T, U, ..., are permissible locutions, and truth functional compounds of statement letters.
- ii. **Withdrawals** ‘No commitment S’ is the locution or withdrawal (retraction) of a statement.

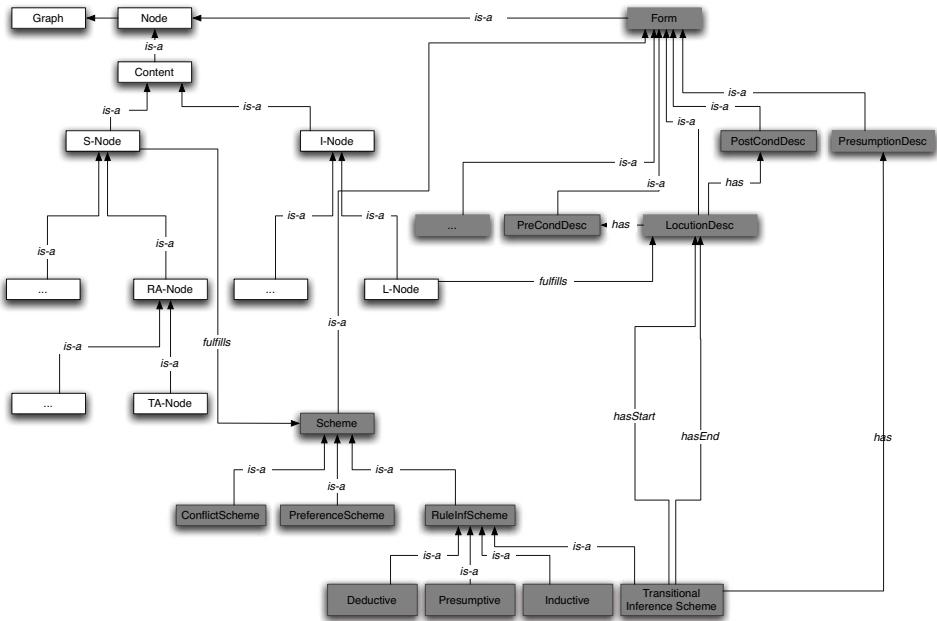


Figure 3. Upper ontology of AIF<sup>+</sup>

- iii. Questions** The question ‘S?’ asks ‘Is it the case that S is true?’
- iv. Challenges** The challenge ‘Why S?’ requests some statement that can serve as a basis in (a possibly defeasible) proof for S.
- v. Critical Attacks** The attack ‘Pose C’ poses the critical question C associated with an argumentation scheme.

### Commitment Rules

- i.** After a player makes a statement, S, it is included in his commitment store.
- ii.** After the withdrawal of S, the statement S is deleted from the speaker’s commitment store.
- iii.** ‘Why S?’ places S in the hearer’s commitment store unless it is already there or unless the hearer immediately retracts his commitment to S.
- iv.** Every statement that is shown by the speaker to be an immediate consequence of statements that are commitments of the hearer via some rule of inference or argumentation scheme A, then becomes a commitment of the hearer’s and is included in the commitment store along with all the assumptions of A.
- v.** No commitment may be withdrawn by the hearer that is shown by the speaker to be an immediate consequence of statements that are previous commitments of the hearer.

## Dialogue Rules

- R1.** Each speaker takes his turn to move by advancing one locution at each turn. A No Commitment locution, however, may accompany a Why-locution as one turn.
- R2.** A question ‘S?’ must be followed by (i) a statement ‘S’, (ii) a statement ‘Not-S’, or (iii) ‘No Commitment S’.
- R3.** ‘Why S?’ must be followed by (i) ‘No commitment S’, or (ii) some statement ‘T’ where S is a consequence of T.
- R4.** After a statement T has been offered in response to a challenge locution, Why S?, then if (S, T) is a substitution instance A of some argumentation scheme of the game, the locution pose(C) is a legal move, where C is a critical question of scheme A appropriately instantiated.
- R5.** After a ‘Pose C’ move, then either (a) if C is an assumption of its argumentation scheme, the move is followed by (i) a statement ‘C’, (ii) a statement ‘not-C’, or (iii) ‘No commitment C’, or (b) if C is an exception to its argumentation scheme, the move is followed by (i) a statement ‘C’ (ii) a statement ‘not-C’ (iii) ‘No commitment C’ , or (iv) ‘Why not-C?’

In the AIF<sup>+</sup> representation of ASD, there are five LocutionDesc nodes which correspond to the five available locutions specified in the ASD locution rules. There are also six explicit transitions, composed from these locutions, which involve particular constraints or presumptions (transitions which are simply inferable from the locutions themselves are captured by a generic, unconstrained transition scheme in much the same way that unspecified inference is captured by a generic rule of inference scheme). For example in ASD a Question locution may be followed by either a Statement or a Withdrawal. In the case of a Question → Statement sequence, the Statement is linked to the preceding Question locution by virtue of the Response transitional inference scheme. When such a *response transition* occurs there is a presumption associated with the transition, that the statement which is uttered in answer to the question actually fulfills the question → answer relationship. The locutions of ASD and the explicit transitions associated with them are illustrated in figure 4 which shows the AIF<sup>+</sup> upper ontology applied to the ASD formal game.

## 6. Dialogue Representation

In the example ASD dialogue provided in [11], there appears the following exchange:

- (L4) Wilma: Well do you remember that “expert” piece that Alf wrote in *South Western Ontario Philosophy Monthly* that said that most Canadian philosophers go to OSSA?
- (L5) Bob: Yes, I remember.
- (L6) Wilma: Well Alf should know, so we can take it that most Canadian philosophers do indeed go.
- (L7) Bob: Yes, but he’d have a biased opinion.
- (L8) Wilma: Why do you think he’s biased?

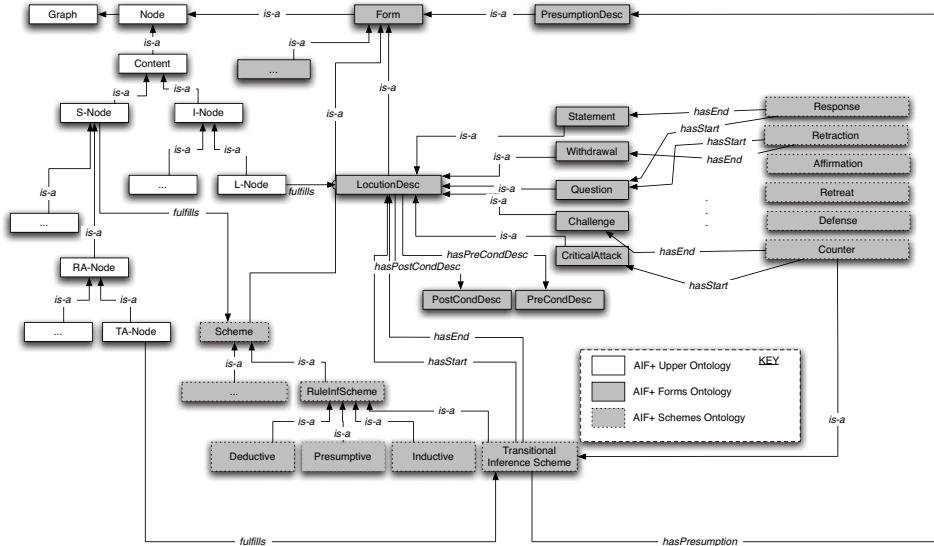


Figure 4. AIF<sup>+</sup> representation of ASD

(L9) Bob: Er, not sure- OK so what if he wasn't biased? So what?

As shown in [11], this may be represented in formal ASD terms as follows-

- (L4) Wilma: (Alf said most Canadian philosophers go to OSSA)? [Question]
- (L5) Bob: (Alf said most Canadian philosophers go to OSSA). [Statement]
- (L6) Wilma: (Most Canadian philosophers go to OSSA). [Statement]
- (L7) Bob: pose(Alf is unbiased). [Critical Attack]
- (L8) Wilma: why(not(Alf is unbiased))? [Challenge]
- (L9) Bob: no-commitment(not(Alf is unbiased)). [Withdrawal]<sup>1</sup>

In this representation, the locutions and their propositional content are easily distinguishable – at (L4), for instance, the locution is “(Alf said most Canadian philosophers go to OSSA)?”, while its propositional content is simply “Alf said most Canadian philosophers go to OSSA”.

The AIF<sup>+</sup> characterisation of this dialogue history is illustrated in figure 5, which falls into two main sections connected by the ‘has-content’ links on the right of the figure. The lower section represents the arguments appealed to during the dialogue – they are conventional AIF material. The upper section represents the actual dialogue itself. The solid-bold-bordered elements represent object-layer entities (capturing the actual data), the grey elements represent intermediate-layer entities (capturing protocols and schemes) and the dashed-bordered elements represent upper-ontology entities (capturing AIF<sup>+</sup> concepts). Some detail is omitted from Figure 5 for clarity - a fuller account of the monologic aspects of the scheme, for example, are given in [9, pp. 18-19].

<sup>1</sup>In [11] L9 is erroneously listed as the statement “(Alf is unbiased).”.

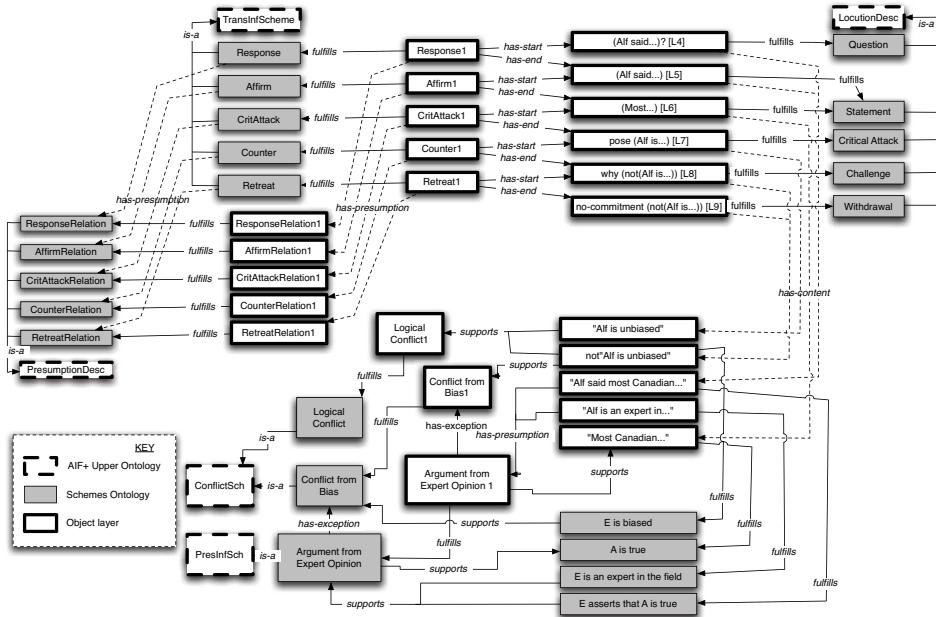


Figure 5. AIF<sup>+</sup> representation of ASD dialogue

## 7. Putting it Together

We have produced a formal description using the Web Ontology Language OWL for both the upper-level ontology shown in figure 3 and the ASD dialogue ontology shown in figure 4. The availability of dialogue game rules in the form of OWL ontologies makes possible the machine processing of dialogue games. In particular, two types of software are under development that make use of these ontologies. The upper-level ontology provides a general framework from which any rule set for any dialogue game can be written by deriving classes from LocutionDesc that describe the particular types of locutions that are permissible in that type of game. The transitional rules between locutions can be formalized by deriving specialized classes from the Transitional Inference Scheme class. Appropriate transition rules are then defined by creating OWL properties corresponding to the ‘hasStart’ and ‘hasEnd’ edges for each transitional inference scheme.

We have given one example of this in the present paper in the form of the ontology for the ASD game. However, this was derived “by hand” here by first sketching out the required diagram on paper and then building a specific OWL ontology from this diagram using Protégé. Although Protégé is relatively easy to use, it is not designed specifically for the production of class hierarchies describing dialogue games. A customized software package that allows the user to build the diagrams for a specific rule set and then generate the OWL automatically from the diagram would clearly be useful here. Such a package would be based on the upper-level ontology described in this paper and available as an OWL file.

The second computational application of the theory described in this paper is in the recording of the locutions in an actual dialogue. Once we have produced the ontology for a specific dialogue game such as ASD and formalized this as in OWL, we can then use the OWL source to execute an instance of a dialogue based on the rules for that game. A software package could provide a graphical interface that allows the user to interact with a computer partner in carrying out a dialogue that is constrained by the rules of the game.

## 8. Conclusions

There have been many examples of generalised machine-representable dialogue protocols and dialogue histories, e.g. [13], [16], but these approaches do not make it easy to identify how the interactions between dialogue moves have effects on structures of argument (i.e. argument\_1), nor how those structures constrain dialogue moves during argument (i.e. argument\_2). Though there are still challenges that the AIF<sup>+</sup> faces in its expressivity and flexibility, we have shown that representing complex protocols that are commitment-based and involve presumptive reasoning forms is straightforward, and that the ways in which those protocols govern or describe dialogue histories is directly analogous to the ways in which schemes govern or describe argument instances. These strong analogies provide ontological parsimony and simplify implementation. This is important because AIF<sup>+</sup> representations, like their AIF predecessors, are far too detailed to create by hand, and the range of software systems will stretch from corpus analysis to agent protocol specification, from Contract Net [15] through agent ludens games [5] to PPD [18]. The success of AIF and AIF<sup>+</sup> will be measured in terms of how well such disparate systems work together.

## References

- [1] C. Chesñevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, and S. Willmott. Towards an argument interchange format. *Knowledge Engineering Review*, 21(4):293–316, 2006.
- [2] P.M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
- [3] M. Kienpointner. How to classify arguments. In F.H. van Eemeren, R. Grootendorst, J.A. Blair, and C.A. Willard, editors, *Argumentation Illuminated*, chapter 15, pages 178–188. SICSAT, 1992.
- [4] J. D. Mackenzie. Question begging in non-cumulative systems. *Journal of Philosophical Logic*, 8:117–133, 1979.
- [5] P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *Journal of Logic, Language and Information*, 11(3):315–334, 2002.
- [6] S. Modgil and J. McGinnis. Towards characterising argumentation based dialogue in the argument interchange format. In *Proceedings of the 4th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS2007)*. Springer Verlag, 2008. to appear.

- [7] D. J. O'Keefe. Two concepts of argument. *The Journal of the American Forensic Association*, 13(3):121–128, 1977.
- [8] H. Van Dyke Parunak. Visualizing agent conversations: Using enhanced dooley graphs for agent design and analysis. In *Proceedings of the 2nd International Conference on Multi-Agent Systems (ICMAS-96)*, pages 275–282. MIT Press, 1996.
- [9] I. Rahwan, F. Zablith, and C. Reed. Laying the foundations for a world wide argument web. *Artificial Intelligence*, 171:897–921, 2007.
- [10] C. Reed. Representing dialogic argumentation. *Knowledge Based Systems*, 19(1):22–31, 2006.
- [11] C. Reed and D. Walton. Argumentation schemes in dialogue. In H.V Hansen, C.W. Tindale, R.H. Johnson, and J.A. Blair, editors, *Dissensus and the Search for Common Ground (Proceedings of OSSA 2007)*, 2007.
- [12] D. Robertson. Multi-agent coordination as distributed logic programming. In B. De moen and V. Lifschitz, editors, *Proceedings of the International Conference on Logic Programming (ICLP-2004)*, pages 416–430. Springer, 2004.
- [13] D. Robertson. A lightweight coordination calculus for agent systems. In *Declarative Agent Languages and Technologies II*, LNCS 3476, pages 183–197. Springer Verlag, 2005.
- [14] J.R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.
- [15] R.G. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113, 1980.
- [16] W3C. Web service choreography interface (wsci) 1.0, 2002.
- [17] D. Walton. *Logical Dialogue Games and Fallacies*. Uni Press of America, 1984.
- [18] D.N. Walton and E.C.W. Krabbe. *Commitment in Dialogue*. SUNY Press, 1995.
- [19] F. Wan and M.P. Singh. Formalizing and achieving multiparty agreements via commitments. In *Proceedings of AAMAS-2005*, pages 770–777, 2005.
- [20] S. Wells. *Formal Dialectical Games in Multi-Agent Argumentation*. PhD thesis, University of Dundee, 2007.

# Heuristics in Argumentation: A Game-Theoretical Investigation<sup>1</sup>

Régis RIVERET<sup>a</sup>, Henry PRAKKEN<sup>b</sup>, Antonino ROTOLI<sup>a</sup>, Giovanni SARTOR<sup>a,c</sup>

<sup>a</sup>*CIRSFID, University of Bologna, Italy*

<sup>b</sup>*Department of Information and Computing Sciences, Utrecht University and Faculty of Law, University of Groningen, The Netherlands*

<sup>c</sup>*European University Institute, Law Department, Florence, Italy*

**Abstract.** This paper provides a game-theoretical investigation on how to determine optimal strategies in dialogue games for argumentation. To make our ideas as widely applicable as possible, we adopt an abstract dialectical setting and model dialogues as extensive games with perfect information where optimal strategies are determined by preferences over outcomes of the disputes. In turn, preferences are specified in terms of expected utility combining the probability of success of arguments with the costs and benefits associated to arguments.

**Keywords.** Argumentation, Game Theory, Expected Utility.

## 1. Introduction

Over the years many dialogue games for argumentation have been proposed to shed light on questions such as which conclusions are (defeasibly) justified, or how procedures for debate and conflict resolution should be structured to arrive at a fair and just outcome.

An issue which has not yet received much attention is the common sense observation that the outcome of a debate does not solely depend on the premises of a case, but also on the strategies that parties in a dispute actually adopt. The problem studied in this paper is how to determine optimal strategies in dialogue games for argumentation. In particular, we will focus on ‘adjudication’ debates. In many debates there are just two participants, who aim to persuade each other to adopt a certain opinion. On the contrary, in adjudication debates, a neutral third party (for example, a judge, a jury or an audience) is involved, who at the end of a debate must decide whether to accept the statements that the opposing parties have made during the debate. In such debates the opposing parties must make estimates about how likely it is that the premises of their arguments will be accepted by the third party, i.e., by the adjudicator. Moreover, we want to take into account that the opposing parties may have non-trivial preferences over the outcome of a debate, so that optimal strategies are determined by two factors: the probability of success of their arguments and the costs/benefits of such arguments.

---

<sup>1</sup>Supported by the EU projects ONE-LEX (Marie Curie Chair), ESTRELLA (IST-2004-027665), and ALIS (IST-2004-027968).

To make our ideas as widely applicable as possible, we formulate our ideas in the abstract dialectical setting of [6, 7]. This allows us to make as few assumptions as possible on the underlying logic and structure of arguments.

A specification of preferences will be provided in terms of expected utility, combining the probability (computed using [8]’s techniques) that an argument, being successful, brings about certain benefits, and the costs of that argument. More precisely, a probability distribution is assumed with respect to the adjudicator’s acceptance of the parties’ statements. This distribution determines the probability of the arguments’ success, which is to be established on the basis also of the probabilities of success of its counterarguments. The probability of success of the argument is then used in combination with the benefits brought about by the argument (if successful) and of its costs in order to predict the expected utility of such an argument. For instance, the benefits brought about by an argument’s success could be the compensation awarded to the successful party. The costs could include the sacrifice of disclosing certain information that the player would have preferred to keep secret, or the expenses required for obtaining the evidence needed for one of its premises (e.g. carrying out expensive laboratory tests). If each argument has an expected utility then each strategy may have a different utility. Game theory allows us to determine the optimal strategies for arguers, that is, the strategies related to the preferred expected utility.

Let us turn to an example (henceforth, “the flat example”) to illustrate our approach. The example is a legal one, since legal disputes are typical examples of adjudication debates.

The proponent *Pro*, John Prator, is the new owner of a flat in Rome. The previous owner sold the flat to John Prator, for the symbolic amount of 10 euros, through a notary deed. The previous owner had signed with Rex Roll, the opponent *Opp*, a rental agreement for the flat, which should apply for two more years. John has an interest in kicking out Rex, since he received an offer from another party, who intends to buy the flat paying 300 000 euros upon the condition that Rex leaves the flat. Rex Roll needs to stay two more years in Rome and refuses to leave the flat, as the rental fee was 500 euros per month, which is a convenient price (other flats in Rome have a rental fee of at least 600 euros per month). Hence, John sues Rex and asks the judge to impose to Rex to leave the flat. We assume that legislation states that any previously signed rental agreement still applies to the new owner when (1) a flat is donated, or (2) flat value is less than 150 000 euros. John’s main argument *A* runs as follows: we do not have a donation since the flat was paid, thus the rental agreement is no longer valid and so Rex Roll has no right to stay. The opponent may present argument *C* that paying a symbolic amount of money indeed configures a case of donation, and John may reply with argument *E* that it is not the case because the property transfer was a sale formalized by a notary deed. Alternatively, Rex may present the argument *B* that the market value of the flat is of 120 000 euros and so the rental agreement is valid, whereas John may reply with *D* saying that he will pay within 10 days 210 000 euros to the previous owner, thus amending the transfer deed in order that it indisputably be a sale concerning a good of a value greater than 150 000 euros. Table 1 provides the probability, costs and benefits for each argument<sup>2</sup> So the problem is the following. According to the analysis of the case, which strategy to adopt?

---

<sup>2</sup>Note that for some arguments, which e.g. counterattack opponent’s attacks, we assume that there is no specific benefits or costs: what they produce is just what is assigned to the main argument *A*. We also assume that, if opponent jointly plays *B* and *C*, this results in the same benefits disjointly produced by *B* and *C*.

This paper is organised as follows. In Section 2 we briefly recall the main notions of [6, 7]'s argument games on which our approach is based. Section 3 provides an interpretation in game theory for such argument games and the definition of optimal strategies. In Section 4, we investigate a specification of the expected utility of a strategy by combining the probability of success of arguments with their associated costs and benefits. In Section 5, related works are briefly discussed. In Section 6, the approach is recapitulated and future investigations are suggested.

## 2. The dialectical setting

The dialectical setting assumed in the paper follows the format of [6, 7]. To specify our dialectical framework we need to provide three sets of assumptions, concerning the logic, the game protocol and the argument games. With regard to the logic we assume the following:

1. Arguments have a finite nonempty set of premises and one conclusion.
2. There is a binary relation of defeat between arguments.

The logical assumptions are complemented with the following requirements concerning the game protocol:

1. An *argument game* is played by two players *Pro* and *Opp*.
2. Informally, a *move* in an argument game is a withdrawal or is an argument that defeats an argument previously moved by the other party (except the first move). Formally, a move is a tuple  $(pl, id, a, t)$  where  $pl$  is the player of the move,  $id$  is the move identifier (a natural number),  $a$  is the argument moved, and  $t$  is the identifier of the move's target. The first and withdrawal moves have a 'dummy' target, to reflect that they do not reply to another move. A withdrawal will be denoted by  $\emptyset$ . Below we will often simply speak of a move as an argument or a withdrawal, leaving the other three elements obvious from the context.
3. Player *Pro* does not repeat moves.
4. Each *turn* of an argument game consists of a withdrawal or a sequence of at most  $m$  arguments such that  $m \geq 1$  ( $m$  is determined by a specific protocol). The first turn consists of a single argument or a withdrawal (i.e. no debate takes place).
5. The turn shifts after a player has made  $m$  moves in a row or earlier if the player to move explicitly indicates that she has ended her turn (which we will leave implicit below).
6. Each move other than the first one defeats its target.
7. If a move is *legal* then it satisfies all preceding conditions. A withdrawal move is always legal. Specific protocols can add further conditions and then turn the 'if' into 'if and only if'.

With regard to argument games we make the following assumptions:

1. A game *terminates* if a player withdraws. If the set of arguments is finite then each game terminates, since the proponent may not repeat arguments.
2. Each game induces a *reply tree*, which consists of the argument moves as nodes and their target relations as links. Note that target (or 'reply') links are, unlike defeat relations from the logic, always unidirectional. Suppose *A* defeats *C*, and

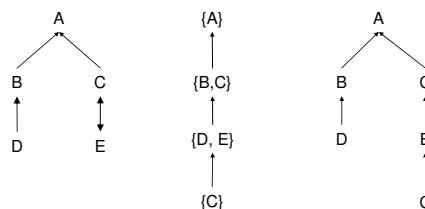
$A$  and  $B$  defeat each other, such that the defeat graph is  $C \leftarrow A \leftrightarrow B$ : this may induce a game tree  $C \leftarrow A \leftarrow B \leftarrow A$ .

3. Reply trees can be labeled as follows: a node is *in* iff all its children are *out*; and a node is *out* iff it has a child that is *in*. (Informally, the leaves of the tree are trivially *in* and then we can work our way upwards through the tree to determine the status of all other nodes.) As proven by [7], this is formally a special case of status assignments to defeat graphs in which each defeat graph has a unique and complete status assignment. If no player (having the possibility to reply with an argument) withdraws, the set of arguments that are *in* corresponds to the unique argument extension in all of the semantics of [3].
4. An argument move  $M$  in a reply tree  $T$  favours *Pro* if  $M$  is *in*; otherwise  $M$  favours *Opp*.
5. A game is *won* by a player if at termination the initial move favours the player.

Given all these assumptions our aim is to define the optimal strategies for *Pro* and *Opp* in argument games. As a matter of fact, we will study this issue for a slightly different kind of game, which is induced by an argument game as defined above. We are in fact interested in optimal *turn* selection (instead of optimal argument selection), so the game to which we will apply the game-theoretical techniques is in fact an argument game consisting of turns, that is, sequences of arguments. So, we work with the following structures:

1. A defeat graph in which the nodes are arguments and the links are defeat relations; which is a declarative representation of a set of available arguments with their defeat relations. The graph is said to be declarative because it does not display in any way whether and when the arguments were stated in a dialogue;
2. A reply tree of a single-move argument game in which the nodes are arguments and the links are reply links;
3. A multi-move argument game which is a sequence of turns by two players *Pro* and *Opp*. Each turn consists of zero or more arguments;
4. A game tree of all possible turn games in which the nodes are turns and the links express their temporal order in a game.

A single terminated argument game based on the defeat graph of the arguments of our example, and its associated reply graph are provided in figure 1.



**Figure 1.** In the middle, a single terminated argument game based on the defeat graph on the left, and its reply graph on the right.

### 3. Game-theoretical model

Game theory deals with the heuristic layer of argumentation dialogue. For an observer, it helps to understand the moves of arguers in a dialogue. For an arguer, it helps to make the right moves by taking into account other arguers' behavior. In order to make the paper self-contained, we provide in the following the relevant game-theoretical notions from [5] slightly adapted to better fit our dialectical setting.

Most basic games presented in the literature are so-called *strategic games* that model situations in which all players' decisions are made simultaneously and each player chooses her plan of action once and for all. Modeling an argumentation dialogue as a strategic game is unsatisfactory because an arguer can plan moves not only at the beginning but also whenever she has to move. In other words, the model of strategic games does not allow arguers to reconsider which arguments to advance after some moves of the other parties. For this reason, we model dialogues as so-called *extensive games* to provide an explicit account of the sequential structure of argumentation. We also assume that arguers are *perfectly informed* about the arguments previously advanced by all other players. Bear in mind that games of perfect information in fact denote cases where no moves are simultaneous. Furthermore, we assume that the set of all arguments and their defeat relations (i.e. the defeat graph) is given in advance, is finite, stays fixed during a game and is known by both players during the game: in game-theoretical terms, we assume a game with *complete* information. Accordingly, an argument game is interpreted as an *extensive games with perfect and complete information*. The players are the opponent and the proponent. An history  $h = (\text{turn}_k)_{k=1 \dots n}$  is a dialogue in an argumentation game, and a terminal history is a terminated dialogue. The function assigning to each non-terminal history a player is the player function of the protocol of the argumentation game. A preference relation is defined over terminal histories. The following defines an extensive game with perfect information adapted to our dialectical setting.

**Definition 1** An extensive argumentation game with perfect information is a 4-tuple  $\langle N, H, P, (\succeq_i) \rangle$  where:

- $N = \{\text{Opp}, \text{Pro}\}$  is a set of arguers, namely the opponent and the proponent;
- $H$  is a set of histories (denoted  $h$ ) which are sequences  $(\text{turn}_k)_{k=1 \dots n}$  of turns  $\text{turn}_k$ . A history  $(\text{turn}_k)_{k=1 \dots K}$  is terminal if it is infinite or if there is no  $\text{turn}_{K+1}$  such that  $(\text{turn}_k)_{k=1 \dots K+1} \in H$ . The set of terminal histories is denoted by  $Z$ ;
- $P$  is a function that assigns to each non-terminal history a member of  $N$  in such a way that the arguers change turns;
- $\succeq_i$  is a preference relation on  $Z$  for each arguer  $i \in N$ .

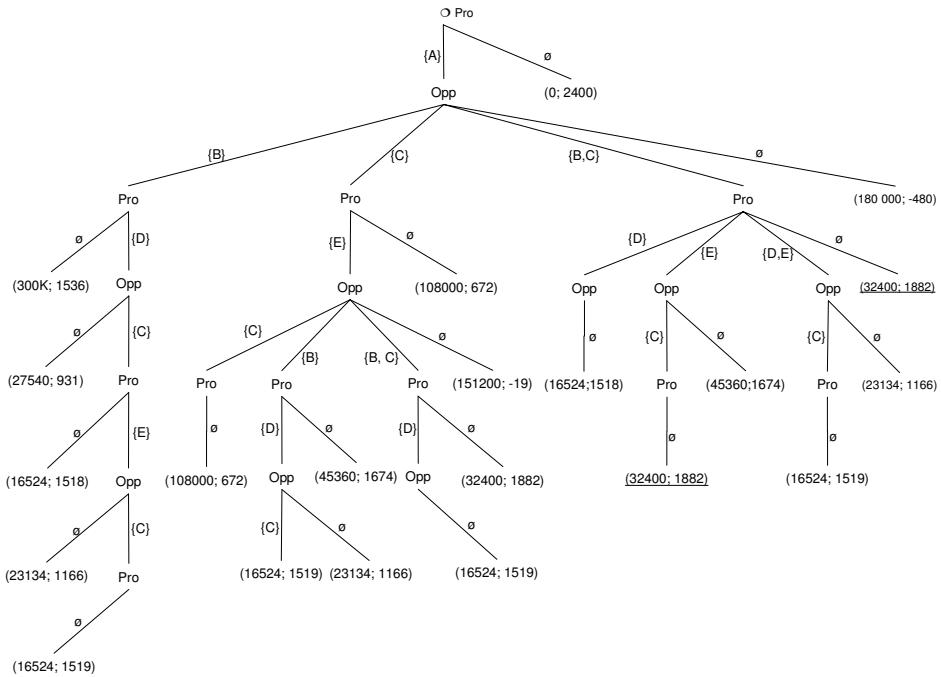
A convenient representation of the argument game tree of our example, interpreted as an extensive game with perfect information, is provided in Figure 2. The set  $H$  of histories consists of the (partial) branches of the tree, and the set  $Z$  of terminated histories consists of the branches ending with the empty turn ( $\emptyset$ ).

We adopt the usual convention that if  $h$  denotes a history and  $t$  turn, then  $(h, t)$  denotes the history that results if history  $h$  is followed by the turn  $t$ . After any nonterminal history  $h$  player  $P(h)$  chooses a move from the set  $M(h) = \{t | (h, t) \in H\}$ .

The strategy of an arguer is defined as the specification of the sequence of arguments chosen by the arguer for every history after which it her turn to move (see [5], p. 92).

Argument	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
Arguer	<i>Pro</i>	<i>Opp</i>	<i>Opp</i>	<i>Pro</i>	<i>Pro</i>
Construction chance	0,6	0,7	0,4	0,3	0,6
$Cost_{Pro}^* / Ben_{Pro}^*$	0/0	0/0	0/0	0/0	0/0
$Cost_{Pro}^{Succ(A)} / Ben_{Pro}^{Succ(A)}$	0/300000	0/0	0/0	210000/0	0/0
$Cost_{Pro}^{\neg Succ(A)} / Ben_{Pro}^{\neg Succ(A)}$	0/0	0/0	0/0	0/0	0/0
$Cost_{Opp}^* / Ben_{Opp}^*$	0/0	0/0	0/0	0/0	0/0
$Cost_{Opp}^{Succ(A)} / Ben_{Opp}^{Succ(A)}$	2400/0	0/0	0/0	0/0	0/0
$Cost_{Opp}^{\neg Succ(A)} / Ben_{Opp}^{\neg Succ(A)}$	0/2400	0/0	0/0	0/0	0/0

**Table 1.** Flat example's arguments. The construction chance and, the different costs and benefits for arguer  $i$  denoted  $Cost_{Pro}^* / Ben_{Pro}^*$ ,  $Cost_i^{Succ(A)} / Ben_i^{Succ(A)}$  and  $Cost_i^{\neg Succ(A)} / Ben_i^{\neg Succ(A)}$  are explained in Section 4.



**Figure 2.** An extensive game tree. The payoffs for the Proponent and Opponent are indicated for each terminal game in terms of expected utility (see Section 4). Underlined payoffs correspond to the perfect equilibria.

**Definition 2** A strategy of arguer  $i \in N$  in an extensive argumentation game with perfect information  $\langle N, H, P, (\succeq_i) \rangle$  is a function that assigns a move  $M(h)$  to each nonterminal history  $h \in H - Z$  for which  $P(h) = i$ .

For each strategy profile  $s = (s_i)_{i \in N}$ , the outcome  $Out(s)$  is the terminal history that results when each player follows her strategy function.

As [5] (p. 93-97) observe, not all Nash equilibria are plausible in extensive game with perfect information; to identify solutions we need the notions of *subgame* and *subgame perfect equilibrium*. For our dialectical setting, such notions are defined as follows.

**Definition 3** The subgame of the extensive argumentation game with perfect infor-

mation  $\Gamma = \langle N, H, P, (\succeq_i) \rangle$  that follows the history  $h$  is the extensive game  $\Gamma(h) = \langle N, H|_h, P|_h, (\succeq_{i|h}) \rangle$ , where

- $H|_h$  is the set of sequences  $h'$  of turns for which  $(h, h') \in H$ , and
- $P|_h$  is defined by  $P|_h(h') = P(h, h')$  for each  $h' \in H|_h$ , and
- $\succeq_{i|h}$  is defined by  $h' \succeq_{i|h} h''$  if and only if  $(h, h') \succeq_i (h, h'')$ .

The strategy  $s_i$  in the subgame  $\Gamma(h)$  is denoted  $s_i|_h$  while the outcome function of  $\Gamma(h)$  is denoted  $Out_h$ . Next we provide an adaptation for argument game of the definition of a subgame perfect equilibrium of an extensive game with perfect information:

**Definition 4** A subgame perfect equilibrium of an extensive argumentation game with perfect information  $\Gamma = \langle N, H, P, (\succeq_i) \rangle$  is a strategy profile  $s^*$  such that for every nonterminal history  $h \in H - Z$  for which  $P(h) = i$ ,  $i \in \{Opp, Pro\}$ , we have:

$$Out_h(s_{Pro}^*|_h, s_{Opp}^*|_h) \succeq_{Opp|h} Out_h(s_{Pro}^*|_h, s_{Opp}|_h)$$

$$Out_h(s_{Pro}^*|_h, s_{Opp}^*|_h) \succeq_{Pro|h} Out_h(s_{Pro}, s_{Opp}^*|_h)$$

for every  $s_{Pro}$  and  $s_{Opp}$  in the subgame  $\Gamma(h)$ .

The preference relation is defined by means of an utility function  $EU_i : Out(s) \rightarrow \mathbb{R}$  such that  $Out(s) \succeq_i Out(s')$  if and only if  $EU_i(Out(s)) \geq EU_i(Out(s'))$ . The equilibria of our example corresponds to the outcomes  $(\{A\}, \{B, C\}, \emptyset)$  and  $(\{A\}, \{B, C\}, \{E\}, \{C\}, \emptyset)$  which both have as associated payoff profile  $(32400, 1882)$ .

The subgame perfect equilibrium can be compiled by using standard backwards induction (see e.g. [5], p. 99). Briefly, backwards induction means that one starts at a player's final decision nodes to see what a player will do there, and then reasons backwards to tell which action is best for the other player.

In a game in which any strategy leads to the same payoff, game theory is trivially useless. Accordingly, game theory is useful in games in which some strategies lead to different outcomes.

Hereafter, we use the following terminology. An arguer is the *expected winner* if, in the tree with the initial argument as root and for each node containing all replies that are legal according to the ‘basic’ argument game, the dialogical status of the root favours that player. An arguer is the *expected loser* if he is not the expected winner. For any arguer  $i$ , the utility function, which assigns value 1 if arguer  $i$  wins and 0 if arguer  $i$  loses, is called the *minimal utility function*. Let us finally call a game protocol *sound* and *complete* if it guarantees that the tree as just defined is traversed in every game. In a game with a sound and complete protocol and a minimal utility function, all the strategies of the expected winner would assure her success at winning the game. Any strategy of the expected loser assures her to lose. Hence, in a game with a sound and complete protocol and minimal utility function, game theory is useless. Accordingly, in a game with a multi-move protocol which is sound and complete, associated to a minimal utility function, game theory is also useless.

A protocol which is not sound and complete may provide strategies leading to different outcomes. For example, in a protocol in which withdrawals are legal (as in the present paper), game theory allows the expected loser to determine strategies which are not leading to her defeat.

Perhaps more interesting, a game in which the utility function is not minimal may also provide strategies leading to different outcomes (even if the protocol is sound and

complete). For example, game theory in any game with a multi-move, sound and complete protocol in association with a utility function compiling costs of moves may help the expected winner to determine the less costly strategy to win. It may be also sometimes better for an arguer to advance fewer arguments than moving more arguments. For instance, the proponent John Partor does not move  $D$  because it is too much costly: after the history  $(\{A\}, \{B, C\})$ , the proponent is better off to argue  $E$  instead of  $\{D, C\}$ .

Changing a protocol may involve changing the perfect equilibria of the game. For example, if the protocol in our case were not multi-move, then all the histories of the extensive game tree in Figure 2 implying a turn with more than one argument would not exist, and the outcome  $(\{A\}, \{B\}, \emptyset)$  would become the unique perfect equilibrium. This shows that game theory is a useful method to test the “fairness” of protocols in terms of their completeness and soundness.

#### 4. Preference specification

In this section we provide a specification of the arguers’ preferences over outcomes in terms of expected utility.

As is well-known, different options are available to calculate expected utility ( $EU$ ) in decision theory and there is an extensive debate in the literature. Well-established trends in decision theory state that decisions should correspond to choosing between risky or uncertain prospects by comparing their expected utility values, which are the weighted sums obtained by adding the utility values of outcomes multiplied by their probabilities. This intuition raises in contemporary debate many questions. For example, what are utility numbers referring to? Are they measured using the same value scale adopted under certainty? Again, Is the weighted sum procedure the only one to be considered? Should it be taken for granted that we rely on probability values, or are there alternative constructions? Finally, we can usually follow two well-known versions of decision theory, namely, Subjective Expected Utility Theory in the case of uncertainty, and von Neumann-Morgenstern Theory in the case of risk.

For the sake of simplicity, we adopt the most classical way to calculate  $EU$  of an act  $X$ , which is the sum of the products of the probabilities and utility’s values for each outcome, formally,  $EU(X) = \sum_{i=1}^n Pr(o_i).u(o_i)$  where  $o_1, \dots, o_n$  are the possible (and mutually exclusive) outcomes of  $X$ .

In our setting, an arguer  $i$  is interested in her expected utility, with regard to the outcome of a strategy profile  $s$  noted  $EU_i(Out(s))$ . The outcome of  $s$ , as we know, is a terminal history, namely, the dialogue resulting from  $s$ . The evaluation of the dialogue depends on the status of its initial node, which results from whether the adjudicator accepts the arguments constituting the dialogue. If according to the adjudicator’s assessment the initial node is *in*, then the proponent of the dialogue has won. If the initial node is *out*, then the opponent has won. For each terminated game associated to strategy profile  $s$ , we have two mutually exclusive outcomes: for each arguer, she can win (the initial move is *in*) or lose (the initial move is *out*). In other words, the initial argument  $A$  is successful or not. Hence, we have:

$$\begin{aligned} EU_i(Out(s)) = & Pr(Succ(A, Out(s))).u_i(Succ(A, Out(s))) \\ & + Pr(\neg Succ(A, Out(s))).u_i(\neg Succ(A, Out(s))) \end{aligned} \quad (1)$$

where  $Pr(Succ(A, Out(s)))$  denotes the probability of success of the initial argument  $A$  w.r.t. the dialogue  $Out(s)$  and  $u_i(Succ(A, Out(s)))$  is the utility value of the success of  $A$  w.r.t. the dialogue  $Out(s)$ .

The probability of success of an argument is intended to mean the probability that the argument is accepted as justified given a knowledge base of which the statements are assigned a probability of acceptance by the adjudicator. The method adopted here is similar to that in [8], but it is slightly adapted to the dialectical setting of the present paper: the probability of success of an argument  $A$  w.r.t. to an argument game is the probability of success of  $A$  w.r.t. the corresponding reply tree. Also, for the sake of simplicity, we assume that no premise of one argument is a conclusion of another argument.<sup>3</sup> The reader is referred to [8] for details and some discussions.

In this setting, the probability of success of an argument depends on two conditions:

1. the probability that the argument's premises are accepted, which we call *construction chance* (the argument will be rejected if the adjudicator refuses to accept one of the argument's premises);
2. the probability that the argument has no valid counterargument, namely no counterargument is able to attack it successfully, which we call the *security chance* (the argument will be rejected if the adjudicator's acceptances imply that there is a valid attacker of the argument).

The construction chance of an argument  $A$ , denoted by  $Pr(Con(A))$ , is given by the probability that the adjudicator accepts all premises  $q_1, \dots, q_n$  of the argument, that is, by the probability of  $q_1^a \wedge \dots \wedge q_n^a$ , where  $q_i^a$  denotes the acceptance of  $q_i$ . On the assumption that the premises of an argument are mutually (statistically) independent,  $Pr(Con(A))$  is the product of the probability of acceptance of all premises in the argument:

$$Pr(Con(A)) = Pr(q_1^a \wedge \dots \wedge q_n^a) = Pr(q_1^a) \times \dots \times Pr(q_n^a) \quad (2)$$

Let us now add the second condition of probability of success of an argument, namely that of not having a successful counterargument. The basic idea is that the chance of success of an argument is diminished to the extent that one of its attackers is going to be successful. Thus, generally, the probability of success of an argument  $A$  is diminished by considering the chances of success of all of its counterarguments, along all possible branches of the reply tree of which the root is given by  $A$ . In the following we shall first consider how to compute the security chance along one line (i.e. a branch) of the reply tree, and then how to compute security chance along multiple lines, that is, in a reply tree.

We first characterise the security chance of an argument  $A_i$  in a branch  $D_n = < A_1, \dots, A_n >$ , intended as the probability that  $A_i$  can really exercise its intended function in the branch.

Since each argument, according to the game protocol, is defeated (and thus prevented from being successful) by its successor, the probability of success of an argument does not only depend on its construction chance, but also on the chance that the argument's successor fails to be successful. Since both these elements need to be present, we have to deal with the probability of a conjunction of the construction of the argument and the failure of its attacker. We therefore define  $Succ(A_i, D_n)$  as  $Con(A_i) \wedge \neg Succ(A_{i+1}, D_n)$ . Since equivalent statements are equally probable we have:

---

<sup>3</sup>In any case, this assumption is reasonable since if some argument's premise is the conclusion of another argument, the two arguments should ideally be combined by making the second argument a subargument of the first, thus turning the premise of the first into an intermediate conclusion.

$$\Pr(\text{Succ}(A_i, D_n)) = \Pr(\text{Con}(A_i)).[1 - \Pr(\text{Succ}(A_{i+1}, D_n) | \text{Con}(A_i))] \quad (3)$$

The chance of success of the last argument  $A_n$  of  $D_n$  is given by its chance of construction:  $\Pr(\text{Succ}(A_n, D_n)) = \Pr(\text{Con}(A_n))$ .

We need to consider the probability of success of an argument taking into account that the argument can have more than one counterargument, and that the probabilities of success of such counterarguments somehow need to be added up (since it is sufficient that one counterargument is successful for the argument to fail). Consider an arbitrary argument  $A$  in a reply tree  $\tau = \langle \tau_1, \dots, \tau_k \rangle$  having counterarguments (children)  $A_1, \dots, A_k$ . The probability that any of these counterarguments  $A_j$  is successful is the probability of the disjunction  $\bigvee_{j=1}^{j=k} \text{Succ}(A_j, \tau_j)$ . Then the probability that  $A$  is successful is given by the probability that  $A$  is constructed times the probability that no such counterargument is successful, that is, that the above disjunction is false. Formally:

$$\begin{aligned} \Pr(\text{Succ}(A, \tau)) &= \Pr(\text{Con}(A) \wedge [\neg(\bigvee_{j=1}^{j=k} \text{Succ}(A_j, \tau_j))] \\ &= \Pr(\text{Con}(A)).\Pr(\bigwedge_{j=1}^{j=k} \neg \text{Succ}(A_j, \tau_j) | \text{Con}(A)) \\ &= \Pr(\text{Con}(A)).\prod_{j=1}^{j=k} \{\Pr(\neg \text{Succ}(A_j, \tau_j) | \text{Con}(A) \wedge \bigwedge_{i=1}^{i=j-1} \neg \text{Succ}(A_i, \tau_i))\} \end{aligned} \quad (4)$$

For example, the security chance of argument  $A$  along the dialogue  $(\{A\}, \{B, C\}, \{D, E\}, \{C\})$  is (assuming that the premises of the arguments are statistically independent):

$$\begin{aligned} \Pr(\text{Succ}(A, (\{A\}, \{B, C\}, \{D, E\}, \{C\}))) &= \Pr(\text{Con}(A)).\{\Pr(\neg \text{Succ}(B, (\{B\}, \{D\})) | \text{Con}(A) \wedge \neg \text{Succ}(C, (\{C\}, \{E\}, \{C\}))) \\ &\quad \cdot \Pr(\neg \text{Succ}(C, (\{C\}, \{E\}, \{C\})) | \text{Con}(A) \wedge \neg \text{Succ}(B, (\{B\}, \{D\})))\} \\ &= \Pr(\text{Con}(A)).\Pr(\neg \text{Succ}(B, (\{B\}, \{D\}))).\Pr(\neg \text{Succ}(C, (\{C\}, \{E\}, \{C\}))) \\ &= \Pr(\text{Con}(A)).[1 - \Pr(\text{Succ}(B, (\{B\}, \{D\})))]. [1 - \Pr(\text{Succ}(C, (\{C\}, \{E\}, \{C\})))] \\ &= 0,1836 \end{aligned} \quad (5)$$

Interestingly, one can demonstrate by induction that the probability of success  $\Pr(\text{Succ}(A_1, D_n))$  of the initial claim  $A_1$  w.r.t. a single argument game  $D_n = \langle t_1, \dots, t_n \rangle$  is bounded in such a way that  $\Pr(\text{Succ}(A_1, t_i)) \geq \Pr(\text{Succ}(A_1, t_j))$  if  $i$  is odd, and  $\Pr(\text{Succ}(A_1, t_i)) \leq \Pr(\text{Succ}(A_1, t_j))$  if  $i$  is even, where  $j \geq i$ . Hence, the probabilities  $\Pr(\text{Succ}(A_1, t_1))$  and  $\Pr(\text{Succ}(A_1, t_2))$  are respectively the highest bound and lowest bound of  $\Pr(\text{Succ}(A_1, t_j))$ . Within these bounds,  $\Pr(\text{Succ}(A_1, t_j))$  oscillates at every move, with the maximum amplitude that decreases with the length of the dialogue.

Next, we focus on the utility values  $u_i(\text{Succ}(A, \text{Out}(s)))$  and  $u_i(\neg \text{Succ}(A, \text{Out}(s)))$  to incorporate costs and benefits of moves. In general, we can distinguish between fixed costs/benefits and costs/benefits dependant upon success. The former ones, in particular, capture costs/benefits independent of the success of the player: for example, some trial expenses for an arguer  $i$  are a fixed cost, since they applies to  $i$  independently of the fact that  $i$  wins or loses. The utility value  $u_i(\text{Succ}(A, \text{Out}(s)))$  is the sum (over the argument  $A_k$  member of  $\text{Out}(s)$ ) of the fixed benefits  $\text{Ben}_i^*(A_k)$  minus the fixed costs  $\text{Cost}_i^*(A_k)$ , plus the sum of benefits  $\text{Ben}_i^{\text{Succ}(A)}(A_k)$  dependant of the success of  $A$  minus the costs  $\text{Cost}_i^{\text{Succ}(A)}(A_k)$  dependant of the success of  $A$ . The utility value  $u_i(\neg \text{Succ}(A, \text{Out}(s)))$  is the sum of the fixed benefits  $\text{Ben}_i^*(A_k)$  minus the fixed costs  $\text{Cost}_i^*(A_k)$ , plus the sum of benefits  $\text{Ben}_i^{\neg \text{Succ}(A)}(A_k)$  dependant of the unsuccess of  $A$  minus the costs  $\text{Cost}_i^{\neg \text{Succ}(A)}(A_k)$  dependant of the unsuccess of  $A$ . Let the set  $\text{Arg}(\text{out}(s)) = \{A_1, \dots, A_K\}$  of arguments constituting  $\text{out}(s)$ , we have formally:

$$\begin{aligned} u_i(\text{Succ}(A, \text{out}(s))) &= \sum_{k=0}^{k=K} (\text{Ben}_i^*(A_k) + \text{Ben}_i^{\text{Succ}(A)}(A_k)) - (\text{Cost}_i^*(A_k) + \text{Cost}_i^{\text{Succ}(A)}(A_k)) \\ u_i(\neg \text{Succ}(A, \text{out}(s))) &= \sum_{k=0}^{k=K} (\text{Ben}_i^*(A_k) + \text{Ben}_i^{\neg \text{Succ}(A)}(A_k)) - (\text{Cost}_i^*(A_k) + \text{Cost}_i^{\neg \text{Succ}(A)}(A_k)) \end{aligned} \quad (6)$$

For example, consider the outcome  $\text{Out}^* = (\{A\}, \{B, C\}, \{E\}, \{C\}, \emptyset)$ , we have  $\text{Arg}(\text{out}^*) = \{A, B, C, E\}$ , and the proponent's expected utility of  $\text{Out}^*$  is  $\text{EU}_{\text{Pro}}(\text{Out}^*) = 0,1836 \times (300\,000 - 210\,000) + (1 - 0,1836) \times 0$ , that is, 32 400. For the sake of simplicity, we have discarded fixed costs and benefits in our example.

As suggested at the end of Section 3, the perfect equilibrium can be compiled using backwards induction (because the reasoning works backwards from outcomes to present decision problems): a player asks herself which of the available final outcomes brings her the highest utility, and chooses the action that starts the chain leading to this outcome.

## 5. Related work

This paper is inspired by [9] in which argumentation is modelled as a game where the payoffs are measured in terms of the probability that the claimed conclusion is, or is not, defeasibly provable, given a history of arguments that have actually been exchanged, and given the probability of acceptance of the factual premises. The probability of a conclusion is calculated using Defeasible Logic, in combination with standard probability calculus. How does [9]'s model compare to the present approach? First, [9]'s game can be called a *theory building game*: during such a game the players jointly build a logical theory by exchanges of arguments, and the outcome of a game is determined by checking whether the topic statement is implied by the end state according to a particular logic. Instead we formulate our ideas in terms of [6, 7]'s notion of the *dialogical status* of a dialogue move. In doing so, we abstract on the underlying logic and structure of arguments, to make our ideas as widely applicable as possible. Second, we retain the idea of uncertainty about statement acceptance, but instead of the probability of success of an argument, [9] proposes to compute the probability that the topic is defeasibly justifiable, where the probability of success of an argument plays no role. By considering the latter probability, the dialogue tree is tracked and that may be a decisive advantage for further investigation in the field of argumentation. Third, the utility function in [9] is reduced to the probability of provability of the claim, whereas the utility function of the present paper account also for costs and benefits of strategies and is thus more fine-grained. Fourth, our protocol is multi-move: this allowed us to illustrate that game theory permits to account for cases in which an arguer is better off to advance few arguments in row.

Another work coupling game theory and argumentation is [1] in which the argumentation techniques of [4] and the game theory approach of [2] are integrated to reach agreement by proposing a trade-off in terms of allocation of numerical utilities representing the importance disputants' place on disputed issues. Roughly, the procedure consists of the following: first, the dialogue techniques are used as an attempt to resolve any existing conflicts; second, the issues which are not resolved are the inputs to a compensation/trade-offs process in order to facilitate resolution of the dispute. If the result of the compensation/trade-offs process is not acceptable by the parties, then they return to the the first step and repeat the whole process recursively until either an agreement or a stalemate is reached. [1]'system is meant to be used in *mediation* procedure setting instead of *litigation*. Litigation causes an argument to be discussed in a law court

so that a judgment can be made which must be accepted by both sides. Instead, mediation is a process by which the participants, together with the assistance of neutral third party, isolate disputed issues in order to reach an agreement. Accordingly, [2]’s game theory investigation is not used for the same aim and cannot apply to litigation. For example, if only one issue needs to be resolved, then suggesting a trade-off is not possible. In the present paper, we are not interested in reaching an agreement by proposing a trade-off in terms of allocation of utilities representing the importance disputants’ place on disputed issues. We aim at determining the optimal strategy of an arguer in dialogue games for argumentation, i.e. an arguer’s sequence of moves which optimises her expected utility.

## 6. Conclusion

The dialectical setting of [6, 7] has been interpreted in game-theoretical terms. This interpretation allowed us to straightforwardly apply game theory, and optimal strategies have been determined accordingly. A specification of preferences over outcomes has been provided in terms of expected utility combining the probability of success of arguments, costs and benefits of arguments. Doing so, we have illustrated that game theory is useful to illuminate diverse aspects of argumentation frameworks.

Future work will focus on assumptions ranging from the specification of the utility function (e.g. assuming non-independent premises) to the game theory modelling (e.g. assuming game with incomplete information, ordinal treatment of subjective utilities). Also, other types of dialogue as negotiation or persuasion could be integrated to the adjudication. Finally, the approach can be implemented into ‘argument assistance systems’ which offer advice and reasons for its advice, to engage into a legal dispute and to choose the optimal strategies.

## References

- [1] E. Bellucci, A. R. Lodder, and J. Zelezniak. Integrating artificial intelligence, argumentation and game theory to develop an online dispute resolution environment. In *ICTAI*, pages 749–754. IEEE Computer Society, 2004.
- [2] E. Bellucci and J. Zelezniak. Representations for decision making support in negotiation. *Journal of Decision Support*, 10(3-4):449–479, 2001.
- [3] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
- [4] A. R. Lodder. *DiaLaw - on legal justification and dialogical models of argumentation*. Iuwer Academic Publishers, Law and Philosophy Library, Volume 42, Dordrecht, 1999.
- [5] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1999.
- [6] H. Prakken. Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese, special issue on New Perspectives in Dialogical Logic*, (127):187–219, 2001.
- [7] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *J. Log. and Comput.*, 15(6):1009–1040, 2005.
- [8] R. Riveret, N. Rotolo, G. Sartor, H. Prakken, and B. Roth. Success chances in argument games: A probabilistic approach to legal disputes. In *Jurix 2007*, To appear, 2007. IOS Press.
- [9] B. Roth, R. Riveret, A. Rotolo, and G. Governatori. Strategic argumentation: a game theoretical investigation. In *ICAIL ’07: Proceedings of the 11th International Conference on Artificial Intelligence and Law*, pages 81–90, New York, NY, USA, 2007. ACM Press.

# Argument Theory Change: Revision Upon Warrant

Nicolás D. ROTSTEIN, Martín O. MOGUILLANSKY,  
Marcelo A. FALAPPA, Alejandro J. GARCÍA and Guillermo R. SIMARI

*Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)*

*Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA)*

*Department of Computer Science and Engineering*

*Universidad Nacional del Sur (UNS), Bahía Blanca, ARGENTINA*

*e-mail: {ndr, mom, maf, ajg, grs}@cs.uns.edu.ar*

**Abstract.** We propose an abstract argumentation theory whose dynamics is captured by the application of belief revision concepts. The theory is deemed as abstract because both the underlying logic for arguments and argumentative semantics remain unspecified. Regarding our approach to argument theory change, we define some basic change operations along with their necessary theoretical elements towards the definition of a warrant-prioritized revision operation. This kind of revision expands the theory by an argument and then applies a contraction ensuring that the added argument can be believed afterwards.

**Keywords.** Formalization of abstract argumentation, Applications

## 1. Introduction & Motivations

In this article, we introduce an abstract theory that captures the dynamics of a proposed argumentation framework through the application of belief revision concepts. Often in the literature, abstract argumentation frameworks [1,2] are built on top of Dung's [3], which does not consider dynamics. Therefore, we define a dynamic abstract argumentation theory including dialectical constraints, and then we present argument revision techniques to describe the fluctuation of the set of active arguments (the ones considered by the inference process of the theory). We claim that our theory is abstract from two standpoints: (1) there is no restriction to any particular representation for arguments nor argumentative semantics, (2) we provide a characterization of the change operators (specially contractions), which is not restricted to a particular implementation.

Belief revision has been applied to an argumentation system in a previous work [4], but in a rather different direction than the proposed here. In [4], a non-prioritized revision is applied in order to insert (partially or totally) a given explanation (a minimal proof) for a sentence to the knowledge base. If the explanation is partially accepted, it is then recognized as an argument (a "defeasible" proof). A closer approach to the one here presented was given in [5], where a non-abstract preliminary investigation on the argument revision matter was introduced.

In our approach, we define expansion, contraction, and revision operators, where the latter can be expressed in terms of the other two, leading to an identity similar to the

one defined by Isaac Levi [6]. Our goal is to define an abstract theory that allows for the introduction of an argument ensuring it can be believed afterwards. This is achieved by applying a revision, that is, an expansion followed by a contraction. The expansion operator is quite straightforward, but (as usual in any model for the theory change) the main complexity relies on the definition of the contraction operator, which allows a wide range of possibilities: from affecting unrestrictedly any number of arguments in the system to keeping this perturbation to a minimum. This choice is up to the minimal change principle followed by the specification of the contraction operation, which also has an indirect impact over the attack relation among arguments.

This paper is organized as follows: first, we introduce the dynamic abstract argumentation framework, then the argument theory change is defined, giving a very brief overview of classic belief revision, and finally the argument change operators are defined, along with an inter-definition between revision and contraction/expansion.

## 2. A Dynamic Abstract Framework with Dialectical Constraints

The framework proposed in this work is formed by arguments, possible ways of interrelating them, and a way of identifying those that are going to be part of the argumentative process. After presenting the dynamic abstract framework, we will give the specific notion of argument used in this article.

**Definition 1 (Dynamic Argumentation Framework)** *A dynamic argumentation framework (DAF)  $\Phi$  is a tuple  $\langle \mathbb{U}, \mathbb{A}, \mathbf{R}, \sqsubseteq \rangle$ , where  $\mathbb{U}$  is a finite set of arguments called universal,  $\mathbb{A} \subseteq \mathbb{U}$  is called the set of active arguments,  $\mathbf{R} \subseteq \mathbb{U} \times \mathbb{U}$  denotes an attack relation between arguments, and  $\sqsubseteq$  is a partial order over  $\mathbb{U}$  called the subargument relation.*

The universal set of arguments  $\mathbb{U}$  characterizes the full set of arguments that could appear in a given domain. At a given instant, the set  $\mathbb{A}$  of active arguments will represent the complete pool of arguments that can be used by the system to make inferences. Note that  $\mathbb{U}$ ,  $\mathbf{R}$ , and  $\sqsubseteq$  are deemed as static, whereas the content of  $\mathbb{A}$  is dynamic, since any change in the system (*i.e.*, its dynamics) is reflected into this set. Having both the universal set of arguments and the subset of the currently active ones allows us to identify the subset of inactive arguments, *i.e.*,  $\mathbb{I} = \mathbb{U} \setminus \mathbb{A}$ . The set of inactive arguments will contain the remainder of arguments (in the universal set) that is not considered by the argumentative process at a specific instant. Later in this article, we will show how inactive arguments can be activated.

In the rest of the article, to refer to attacks between arguments, we will simply use the word “attacks” or the notation  $\mathcal{A}_1 \mathbf{R} \mathcal{A}_2$ , which means that  $\mathcal{A}_1$  attacks  $\mathcal{A}_2$ , or equivalently that  $\mathcal{A}_1$  is a defeater for  $\mathcal{A}_2$ . As in [7], the symbol  $\sqsubseteq$  denotes subargument relation:  $\mathcal{A} \sqsubseteq \mathcal{B}$  means that  $\mathcal{A}$  is a subargument of  $\mathcal{B}$  and  $\mathcal{B}$  is a superargument of  $\mathcal{A}$ . Subarguments are arguments; therefore, every subargument belongs to  $\mathbb{U}$ , and they are a (distinguishable) part of the arguments they support. In this work, we will use the subargument concept to be able to eliminate some part of a given argument. The reason for this will be clear in the next section. We will refer to a proper subargument  $\mathcal{A}_i$  of an argument  $\mathcal{A}$  as  $\mathcal{A}_i \sqsubset \mathcal{A}$ , meaning that  $\mathcal{A}_i \sqsubseteq \mathcal{A}$ , but  $\mathcal{A}_i \neq \mathcal{A}$ . Finally, since the subargument relation is a partial order, it meets the properties of transitivity, antisymmetry, and reflexivity.

**Definition 2 (Argument)** An **argument** is a set of interrelated pieces of knowledge supporting a claim from evidence and satisfying: **Self-Consistency**:  $\mathcal{A}$  is self-consistent wrt.  $\mathbf{R}$  iff there are no  $\mathcal{A}_i \sqsubseteq \mathcal{A}$ ,  $\mathcal{A}_j \sqsubseteq \mathcal{A}$  such that  $\mathcal{A}_i \mathbf{R} \mathcal{A}_j$  nor  $\mathcal{A}_j \mathbf{R} \mathcal{A}_i$ . **Minimality**:  $\mathcal{A}$  is minimal iff  $\mathcal{A}$  supports  $\alpha$  and there is no  $\mathcal{A}_i \sqsubset \mathcal{A}$  such that  $\mathcal{A}_i$  supports  $\alpha$ .

**Definition 3 (Atomic Argument)** Let  $\langle \mathbb{U}, \mathbb{A}, \mathbf{R}, \sqsubseteq \rangle$  be a DAF. An argument  $\mathcal{A} \in \mathbb{U}$  is **atomic** iff there is no  $\mathcal{B} \in \mathbb{U}$  such that  $\mathcal{B} \sqsubset \mathcal{A}$ .

We will refer as *regular arguments* to those defined in the usual sense, that is, as a reason that supports a claim upon available evidence. We will consider *evidence* to be an active regular atomic argument, i.e., a piece of evidence is an argument by itself. This association turns out to be quite natural, since pieces of evidence can be thought as indivisible and (self) conclusive. Note that an inactive regular atomic argument will not be considered as evidence. Atomic arguments will also allow us to identify the building blocks of an argument, i.e., its minimal portions, as will be clear in Example 1.

Besides regular arguments, we will identify *potential arguments* as a supporting structure that is incomplete due to a lack of evidence (analogous to the concept introduced in [8]). Thus, although potential arguments have an associated claim, they cannot derive it by themselves. When evidence is not available, potential arguments will need claims from other arguments to be able to reach their own. Therefore, the set  $\mathbb{U}$  can be seen as containing arguments that are not only active or inactive, but can be also regular or potential, and even atomic or not, thus dividing  $\mathbb{U}$  in eight classes of argument. The need for three levels of classification will be clear in Section 3. For the definition of a potential argument the following functions are necessary:

**Atomic:**  $\mu : \mathbb{U} \rightarrow \mathcal{P}(\mathbb{U})$  is such that  $\mu(\mathcal{A}) = \{\mathcal{A}_i | \mathcal{A}_i \text{ is an atomic argument, } \mathcal{A}_i \sqsubseteq \mathcal{A}\}$

**Completion:**  $\chi : \mathbb{U} \rightarrow \mathbb{U}$  where  $\chi(\mathcal{A})$  is a regular argument  $\mathcal{B}$  composed only by atomic arguments, such that  $\mu(\mathcal{A}) \subset \mu(\mathcal{B})$ , and  $\mu(\mathcal{B}) \setminus \mu(\mathcal{A})$  is a non-empty set of regular atomic arguments.

There are multiple ways of completing any given argument, but the completion function  $\chi$  gives the only one composed by the atomic arguments of  $\mathcal{A}$  along with the necessary regular atomic arguments to support the claim.

**Definition 4 (Potential Argument)**  $\mathcal{A}$  is a **potential argument** for  $\alpha$  iff  $\mathcal{A}$  does not support  $\alpha$  and the completion  $\chi(\mathcal{A})$  is an argument for  $\alpha$  (i.e., it satisfies minimality and self-consistency).

In order for a potential argument to support its claim, other arguments should provide their own claims as a replacement for the missing evidence. Despite claims supported by an argument can be used when no evidence is available<sup>1</sup>, they clearly differ in semantics: evidence is indisputable, whereas claims supported by an argument could be defeated. The notions of minimality and self-consistency indirectly apply to potential arguments, since their completion is a regular argument.

We will identify the set of regular arguments as **Reg** and the set of potential arguments as **Pot**. When convenient, these sets will be subscripted with  $\mathbb{A}$  or  $\mathbb{I}$ , according to their nature of active or inactive. In what follows, we will use a graphical notation, intu-

---

<sup>1</sup>Note that a claim does not appear by itself, but it is always supported by an argument.

itive in nature, to depict regular and potential arguments: both will be represented as triangles, but potential arguments will contain a black “slot” in their base. The position of subarguments within an argument will suggest that upper potential subarguments need the claim of those placed at lower positions in order to reach a claim. This is shown in Figures 1 and 2, where subarguments at the base are required to reach a claim in order for the subargument at the top to reach its own.

For the following example only, we will assume a structure for arguments by using (propositional) logic programming. This will be useful to understand the graphical representation of arguments and subarguments and their condition of regular or potential.

### Example 1

Consider an argument  $\mathcal{A} = \{c \leftarrow b, b \leftarrow a, a\}$ , then we will say that  $\mathcal{A}$  supports  $c$ , therefore  $\mathcal{A} \in \text{Reg}$ . Moreover, the subargument  $\mathcal{A}_1 = \{b \leftarrow a, a\}$  also belongs to  $\text{Reg}$  provided it supports  $b$ , but then the subargument  $\mathcal{A}_2 = \{c \leftarrow b\}$  belongs to  $\text{Pot}$  given it cannot reach its claim  $c$  by itself. Furthermore, two subarguments  $\mathcal{A}_{11}, \mathcal{A}_{12} \sqsubseteq \mathcal{A}_1$  are such that  $\mathcal{A}_{11} = \{a\}$  and  $\mathcal{A}_{12} = \{b \leftarrow a\}$ , where the latter belongs to  $\text{Pot}$  and the former to  $\text{Reg}$ . This configuration is depicted in the figure. Notice that this graphical notation can replace the logical representation for arguments. From now on, we are going to rely on the graphical representation in order to make a complete abstraction over any underlying logic of arguments.

The graphical representation allows us to recognize a particular configuration of subarguments: if a potential subargument is placed at the bottom of an argument triangle, then this argument is potential. For instance, in Figure 1 we have that argument  $\mathcal{A}$  contains arguments  $\mathcal{A}_1, \mathcal{A}_2$  and  $\mathcal{A}_3$ , where  $\mathcal{A}_1$  is a potential argument that is fed by the claims of  $\mathcal{A}_2$  and  $\mathcal{A}_3$ . Thus, since  $\mathcal{A}_3$  is a potential subargument of  $\mathcal{A}$  (placed at its base), then  $\mathcal{A}$  is also potential. In opposition to this, in Figure 2 is shown a similar case, in which both arguments at the base of the triangle ( $\mathcal{B}_2$  and  $\mathcal{B}_3$ ) are not potential, and therefore the whole argument  $\mathcal{B}$  is not potential.

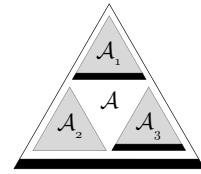


Figure 1.  $\mathcal{A} \in \text{Pot}$

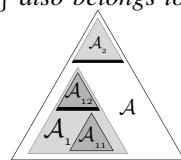
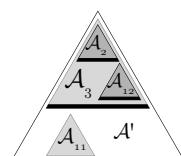


Figure 2.  $\mathcal{B} \in \text{Reg}$

**Definition 5 (Equisubstructural Arguments)** Let  $\mu$  be the Atomic function. Two arguments  $\mathcal{A}_1$  and  $\mathcal{A}_2$  are said to be **equisubstructural** iff  $\mu(\mathcal{A}_1) = \mu(\mathcal{A}_2)$ .

### Example 2

We can give an alternative representation  $\mathcal{A}'$  for argument  $\mathcal{A}$  in Example 1. Since the sets of atomic arguments (i.e.,  $\mathcal{A}_{11}, \mathcal{A}_{12}, \mathcal{A}_2$ ) of  $\mathcal{A}$  and  $\mathcal{A}'$  are the same,  $\mathcal{A}$  and  $\mathcal{A}'$  are equisubstructural. Note that the order from bottom to top of the atomic arguments is the same in both cases, as would be expected.



Having more than one representation for the same argument could be a problem; in those situations, equisubstructurality would allow us to identify the class of equisubstructural arguments from which just one should be used.

As stated above, arguments can be either active or inactive. The interaction between arguments and subarguments regarding activeness will be made clear by the **activeness**

**propagation** principle, which defines the dynamics of the system. A group of arguments being active determines that an argument containing them (exclusively) is going to be active. Furthermore, a single argument becoming active makes all of its subarguments to become active. This also works the other way around: if a subargument  $\mathcal{A}_i$  of an argument  $\mathcal{A}$  is set inactive, then every superargument of  $\mathcal{A}_i$  is set inactive, including  $\mathcal{A}$ . Moreover, the inactiveness of  $\mathcal{A}_i$  means that at least one subargument of it is inactive. Activeness propagation is formally defined as follows:

**(Activeness Propagation)**  $\mathcal{A} \in \mathbb{A}$  iff for every  $\mathcal{A}_i \sqsubseteq \mathcal{A}$  we have that  $\mathcal{A}_i \in \mathbb{A}$ .

As a basis for our analysis we will use a DAF enriched with the notion of *dialectical constraint*. The resulting extended framework will be called a *dynamic argumentation theory*. The following definitions are an extended version of those in [9].

**Definition 6 (Argumentation Line)** Let  $\Phi$  be a DAF. An *argumentation line*  $\lambda$  in  $\Phi$  is any finite sequence of arguments  $[\mathcal{A}_1, \dots, \mathcal{A}_n]$  such that  $\mathcal{A}_i \mathbf{R} \mathcal{A}_{i-1}$ , for  $1 < i \leq n$ . If  $\mathcal{A}_1$  is the first element in  $\lambda$ , we will also say that  $\lambda$  is *rooted in*  $\mathcal{A}_1$ . The *upper segment* of  $\lambda$  wrt.  $\mathcal{A}_i$ , is defined as  $\lambda^\uparrow(\mathcal{A}_{i>1}) = [\mathcal{A}_1, \dots, \mathcal{A}_{i-1}]$ , and  $\lambda^\uparrow(\mathcal{A}_1)$  does not exist.

**Definition 7 (Set of Interference (Supporting) Arguments)** Let  $\lambda$  be an argumentation line, then the *set of interference* (resp., *supporting*) arguments  $\lambda^-$  (resp.,  $\lambda^+$ ) of  $\lambda$  is the set containing all the arguments placed on even (resp., odd) positions in  $\lambda$ .

We will write  $\mathfrak{L}\mathbf{i}\mathbf{n}\mathbf{e}\mathfrak{s}_\Phi$  to denote the set of all possible argumentation lines regarding the arguments in  $\Phi$ . These lines define a domain onto which different constraints can be defined. As such constraints are related to sequences which resemble an argumentation dialogue between two parties, we call them *dialectical constraints*. Formally:

**Definition 8 (Dialectical Constraint)** Let  $\Phi$  be a DAF. A *dialectical constraint*  $\mathbf{C}$  in the context of  $\Phi$  is any function  $\mathbf{C} : \mathfrak{L}\mathbf{i}\mathbf{n}\mathbf{e}\mathfrak{s}_\Phi \rightarrow \{\text{True}, \text{False}\}$ . A given argument sequence  $\lambda \in \mathfrak{L}\mathbf{i}\mathbf{n}\mathbf{e}\mathfrak{s}_\Phi$  satisfies  $\mathbf{C}$  in  $\Phi$  when  $\mathbf{C}(\lambda) = \text{True}$ .

**Definition 9 (Dynamic Argumentation Theory)** A *dynamic argumentation theory* (DAT)  $T$  is a pair  $(\Phi, \mathbf{DC})$ , where  $\Phi$  is a DAF satisfying activeness propagation, and  $\mathbf{DC} = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_k\}$  is a finite (possibly empty) set of dialectical constraints.

**Definition 10 (Acceptable Argumentation Line)** Given a DAT  $T = (\Phi, \mathbf{DC})$ , an argumentation line  $\lambda$  is *acceptable* wrt.  $T$  iff  $\lambda$  satisfies every  $\mathbf{C}_i \in \mathbf{DC}$ , and every  $\mathcal{B} \in \lambda$  is *active regular*.

In what follows, we will assume that the notion of acceptability imposed by dialectical constraints is such that if  $\lambda$  is acceptable wrt. a DAT  $T = (\Phi, \mathbf{DC})$ , then any subsequence of  $\lambda$  is also acceptable. We also assume a dialectical constraint that avoids the construction of circular argumentation lines, thus no line will contain two or more equistructural arguments.

**Definition 11 (Bundle Set)** Given a DAT  $T$ , a set  $S = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$  of argumentation lines rooted in a given argument  $\mathcal{A}$ , denoted  $S_{\mathcal{A}}$ , is called a *bundle set* wrt.  $T$  iff there is no pair  $\lambda_i, \lambda_j \in S_{\mathcal{A}}$  such that  $\lambda_j$  is a subsequence of  $\lambda_i$ .

**Definition 12 (Dialectical Tree)** Let  $T$  be a DAT, and let  $\mathcal{A}$  be an argument in  $T$  and let  $S_{\mathcal{A}} = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$  be a bundle set. The **dialectical tree** rooted in  $\mathcal{A}$  based on  $S_{\mathcal{A}}$  (denoted  $\mathcal{T}_{\mathcal{A}}$ ) is a tree-like structure defined as follows:

1. The root node of  $\mathcal{T}_{\mathcal{A}}$  is  $\mathcal{A}$ .
  2. Let  $F = \{\text{tail}(\lambda), \text{for every } \lambda \in S_{\mathcal{A}}\}$ , and  $H = \{\text{head}(\lambda), \text{for every } \lambda \in F\}$ .<sup>2</sup> If  $H = \emptyset$  then  $\mathcal{T}_{\mathcal{A}}$  has no subtrees. Otherwise, if  $H = \{\mathcal{B}_1, \dots, \mathcal{B}_n\}$ , then for every  $\mathcal{B}_i \in H$ , we define:  $\text{getBundle}(\mathcal{B}_i) = \{\lambda \in F \mid \text{head}(\lambda) = \mathcal{B}_i\}$
- We put  $\mathcal{T}_{\mathcal{B}_i}$  as the immediate subtree or  $\mathcal{A}$  based on  $\text{getBundle}(\mathcal{B}_i)$ .

We will denote  $\mathbf{Tree}_T$  to the family of all possible dialectical trees in the DAT  $T$ .

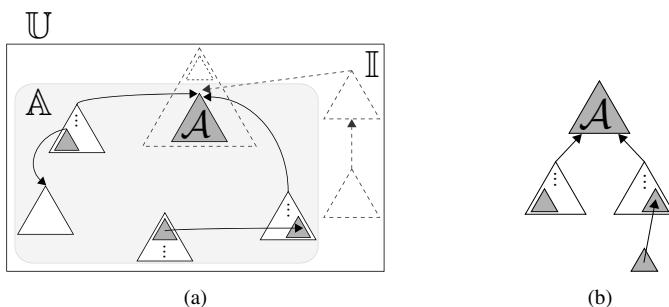
Acceptable dialectical trees are a subclass of dialectical trees that contain only acceptable argumentation lines. In the sequel, we will just write “dialectical trees” to refer to acceptable dialectical trees, unless stated otherwise. Acceptable dialectical trees allow to determine whether the root node of the tree is to be accepted (ultimately *undefeated*) or rejected (ultimately *defeated*) as a rationally justified belief. A *marking function* provides a definition of such acceptance criterion. Formally:

**Definition 13 (Marking criterion)** Let  $T$  be a DAT. A **marking criterion** for  $T$  is a function  $\text{Mark} : \mathbf{Tree}_T \rightarrow \{D, U\}$ . We will write  $\text{Mark}(\mathcal{T}_{\mathcal{A}}) = U$  (resp.  $\text{Mark}(\mathcal{T}_{\mathcal{A}}) = D$ ) to denote that the root node  $\mathcal{A}$  of  $\mathcal{T}_{\mathcal{A}}$  is marked as *undefeated* (resp. *defeated*).

**Definition 14 (Warrant)** Let  $T$  be a DAT and  $\text{Mark}$  a marking criterion for  $T$ . An active regular argument  $\mathcal{A}$  is a **warranted argument** (or just **warrant**) wrt. a marking criterion  $\text{Mark}$  in  $T$  iff the dialectical tree  $\mathcal{T}_{\mathcal{A}}$  is such that  $\text{Mark}(\mathcal{T}_{\mathcal{A}}) = U$ .

### Example 3

The digraph of arguments in Figure 3(a) describes a DAF, where the nodes are arguments and the arcs denote the attack relation with the arrowhead pointing to the argument under attack. The set  $\mathbb{A}$  of active arguments is shown, along with the universal  $\mathbb{U}$ , and the set  $\mathbb{I}$  of inactive arguments, which are illustrated as dashed triangles. Subarguments were drawn following the aforementioned convention. Note that the superargument of  $\mathcal{A}$  is inactive because it has an inactive subargument.



**Figure 3.** (a) Dynamic argumentation framework example (b) Tree spanning from  $\mathcal{A}$

Figure 3(b) shows a dialectical tree spanning the graph from argument  $\mathcal{A}$ . Observe that despite an attack occurs between an inactive and an active argument, inactive ar-

<sup>2</sup>The functions  $\text{head}(\lambda)$  and  $\text{tail}(\lambda)$  have the usual meaning in list processing.

gments are not considered when analyzing the tree. The marking of this dialectical tree would allow us to determine if the root argument is warranted. Consider a marking function where each node of the tree is undefeated if either it is a leaf or all of its defeators are defeated. With such a marking, the root node would be defeated. This status could be changed if we deactivate either the root's left defeater or both.

### 3. An Approach to an Argument Theory Change

We will briefly introduce some of the basic concepts of the belief revision theory [10]. Classic operations in the theory change, as those specified in the AGM model [11], are known as expansions, contractions, and revisions. An *expansion* adds a new belief to the epistemic state without guaranteeing its consistency after the operation. A *contraction* eliminates a belief from the epistemic state and some beliefs that make possible its deduction. The sentences to eliminate might represent the *minimal change* on the epistemic state. Finally, a *revision* inserts a sentence into the epistemic state, guaranteeing consistency if the input sentence was consistent. Hence, a revision adds a new belief possibly eliminating others to avoid inconsistencies. The latter change operation has been defined through the Levi Identity [6], which is a composition of sub-operations that ensures consistency by contracting the negation of the sentence at issue, and therefore by expanding it to the resulting knowledge base.

Regarding an argument change theory, a useful kind of revision would be to add an argument to a theory in such a way that this argument ends up being warranted. Therefore, in the rest of the article, we explore a contraction operator that will allow us to define a revision operator that follows the desired behavior. Firstly, we will introduce the basic theoretical elements required to modify a dialectical tree and turn the marking of its root argument to warranted. Then, we will define these three operations: (1) *Argument expansion*, which activates an argument; (2) *Non-warrant argument contraction*, which deactivates arguments in a particular tree looking to warrant its root; (3) *Warrant-prioritized argument revision* (WPA revision), defined from the previous operations.

#### 3.1. Basic Theoretical Elements for Argument Theory Change

We need to characterize the kind of argumentation lines that actually affect the status of the root argument. We will call these lines *attacking lines*, and will be those over which the argument selection and then the argument incision are going to be applied. Although the main idea is to turn every attacking line into non-attacking, the correctness of the revision must not depend on whether it is possible to determine which lines are attacking; that is, if selections and incisions are applied to every line in the tree, the revision should remain correct. The condition of attacking for a given argumentation line is strictly dependant on the adopted marking function.

**Definition 15 (Set of Attacking Lines)** Given a DAT  $T$  and the dialectical tree  $\mathcal{T}_A$  based on the bundle set  $S_A$ , the set of **attacking lines**  $\text{Att}_A$  over  $A$  is the minimal subset of  $S_A$  such that  $S'_A = S_A \setminus \text{Att}_A$  is the bundle set for  $A$  in a hypothetical DAT  $T'$ , where the dialectical tree based on  $S'_A$  warrants  $A$ .

**Remark 1 (Notational Shortcuts)** In the rest of the article, we will work on a DAT  $T = (\Phi, \text{DC})$ ,  $\Phi = \langle \mathbb{U}, \mathbb{A}, \mathbf{R}, \sqsubseteq \rangle$  with an active regular argument  $\mathcal{A}$  as the root of a dialectical tree  $T_{\mathcal{A}}$  (from  $\text{Tree}_T$ ) based on a bundle set  $S_{\mathcal{A}} \subseteq \text{Lines}_{\Phi}$ , where its (acceptable) argumentation lines  $\lambda_i \in S_{\mathcal{A}}$  are to be associated to  $T_{\mathcal{A}}$  by the notation  $\lambda_i \in T_{\mathcal{A}}$ . Arguments will be noted only as  $\mathcal{A}$ ,  $\mathcal{B}$ , and  $\mathcal{C}$ , being  $\mathcal{A}$  always the root of  $T_{\mathcal{A}}$ , and  $\mathcal{B}$  and  $\mathcal{C}$ , inner nodes in that tree, i.e.,  $\mathcal{A}, \mathcal{B}, \mathcal{C} \in T_{\mathcal{A}}$ . Finally,  $\mathcal{B}_i^+$  (resp.  $\mathcal{B}_i^-$ ) will mean that  $\mathcal{B} \in \lambda_i^+$  (resp.  $\mathcal{B} \in \lambda_i^-$ ).

**Definition 16 (Argument Selection Function “ $\gamma$ ”)** An **argument selection function**  $\gamma : \text{Lines}_{\Phi} \rightarrow \mathbb{U}$  is applied to every attacking line  $\lambda_i \in \text{Att}_{\mathcal{A}}$  in such a way that  $\gamma(\lambda_i) = \mathcal{B}_i^-$  and  $\lambda_i^+(\mathcal{B}_i^-) \notin \text{Att}_{\mathcal{A}}$ . In what follows, we will refer to the selected argument  $\mathcal{B}_i^-$  just as  $\Psi_i$ .

It is reasonable to require the argument selection function to return an interference argument, since these arguments are the ones that contradict the root. The condition of attacking of a given line should only depend on interference arguments –deactivating a supporting argument should not turn an attacking line into a non-attacking line. Furthermore, the deactivation of an interference argument from a non-attacking line should not turn this line into an attacking line.

**Definition 17 (Argument Incision Function “ $\sigma$ ”)** A function  $\sigma : \mathbb{U} \rightarrow \mathcal{P}(\mathbb{U})$  is an **argument incision function** iff  $\emptyset \subset \sigma(\Psi_i) \subseteq \mu(\Psi_i)$ .

Incisions should be guided by an “intra-argument” criterion. For instance, they could be defined following some epistemic entrenchment method, namely, evidence might be considered more important than any other subargument. Therefore, it would be preserved from being cut off, unless it is not possible. The way arguments are incised should be defined by an internal selection function, which is out of the scope of this paper. Sometimes incisions will affect more arguments than the one being incised. In order to identify this situation, we introduce the notion of *collateral incision*; formally:

**Definition 18 (Collateral Incision)** A **collateral incision** over  $\mathcal{B}_j$  is defined as  $\sigma(\Psi_i) \cap \mu(\mathcal{B}_j) \neq \emptyset$ . If  $\sigma(\Psi_i) \cap \mu(\mathcal{C}_j) = \emptyset$  for every  $\mathcal{C}_j \in \lambda_j^+(\mathcal{B}_j)$ , we will say that  $\sigma(\Psi_i)^{(\mathcal{B}_j)} = \sigma(\Psi_i) \cap \mu(\mathcal{B}_j)$  is the **uppermost collateral incision**.

Collateral incisions bring about some drawbacks: supporting arguments could be involuntarily deactivated, which might turn a non-attacking line into an attacking line. Besides, as said before, it is not reasonable to think that if the supporting argument belonged to an attacking line, its deactivation would change the line’s status. Moreover, although a collaterally incised interference argument does not turn lines into attacking, it would also be an unnecessary incision. Therefore, it is desirable to select arguments in which any incision would never result in a collateral incision to other arguments. This is captured by the *cautiousness* property:

$$(\text{Cautiousness}) \quad \mu(\Psi) \cap \mu(\mathcal{B}) = \emptyset, \text{ for any } \mathcal{B}$$

**Definition 19 (Cautious and Non-Cautious Selections)** A selection  $\Psi$  is identified as **cautious** iff it verifies **cautiousness**; otherwise, it is identified as **non-cautious**.

Sometimes cautiousness may not be satisfied. In such a case, when a non-cautious selection is unavoidable, the incision in it should avoid any collateral incisions over any argument. However, this situation may not be always prevented and should be properly addressed. These difficulties are captured by the following principle:

**(Preservation)** If  $\sigma(\Psi_i)^{(\mathcal{B}_j)} \neq \emptyset$  then  
exists  $\lambda_j^\uparrow(\mathcal{B}_j)$  and  $(\Psi_j \in \lambda_j^\uparrow(\mathcal{B}_j))$  iff  $\lambda_j^\uparrow(\mathcal{B}_j) \in \text{Att}_{\mathcal{A}}$ ,  
for any  $\mathcal{B}_j$

This principle is illustrated in Figure 4. When an incision  $\sigma(\Psi_i)$  in the  $i^{th}$  dialectical line (the left branch in Figure 4) results in an uppermost collateral incision  $\sigma(\Psi_i)^{(\mathcal{B}_j)}$  over argument  $\mathcal{B}_j$  in the  $j^{th}$  dialectical line (right branch), it must be ensured that the selection  $\Psi_j$  in the  $j^{th}$  line is performed over the upper segment  $\lambda_j^\uparrow(\mathcal{B}_j)$ . This selection is only performed if  $\lambda_j^\uparrow(\mathcal{B}_j)$  is an attacking line. Finally, note that if  $\mathcal{B}_j$  is the root node, then there is no upper segment for it.

In the case of the antecedent of the preservation principle being false (when there is no collateral incision over any argument  $\mathcal{B}_j$  in any  $j^{th}$  line) the validity of the preservation principle is not threatened. This particular case may be referred as:

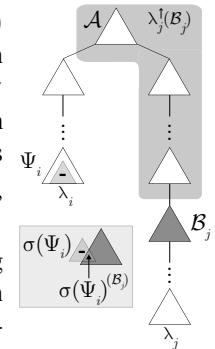


Figure 4. Preservation

**(Strict-Preservation)**  $\sigma(\Psi)^{(\mathcal{B})} = \emptyset$ , for any  $\mathcal{B}$

An incision satisfying strict-preservation ensures no argument is collaterally incised, although this principle cannot be always verified. The following two propositions address the relation between cautiousness and strict-preservation. Proposition 1.1 states that, when a selection is cautious, there is no overlapping with any argument; therefore, the incision over that selection verifies strict-preservation. However, a non-cautious selection could verify strict-preservation if, even though it overlaps with some argument, the incision over that selection is performed outside this overlapping. In this case, there is no collateral incision and strict-preservation holds (Proposition 1.2). Achieving strict-preservation regardless cautiousness may be also a desirable property.

### Proposition 1<sup>3</sup>

- (1) If  $\Psi_i$  is cautious, then  $\sigma(\Psi_i)$  is strict-preserving.
- (2) If  $\Psi_i$  is non-cautious and there exists  $\mathcal{C} \in \mu(\Psi_i)$  such that  $\mathcal{C} \not\sqsubset \mathcal{B}_j$  (for every  $\mathcal{B}_j$ ), then there exists some  $\sigma(\Psi_i)$  such that it is strict-preserving.

Regarding collateral incisions, it is paramount to preserve the root argument  $\mathcal{A}$  as active. In order to achieve this, no collateral incision should affect any subargument of  $\mathcal{A}$ ; otherwise, it would be impossible to warrant it.

**(Root-Preservation)**  $\sigma(\Psi)^{(\mathcal{A})} = \emptyset$

**Root-preservation** is a particular case of **strict-preservation**, where the argument  $\mathcal{B}$  is the root argument  $\mathcal{A}$ . Since  $\lambda_j^\uparrow(\mathcal{A})$  does not exist, the consequent of this instance

<sup>3</sup>Formal proofs were omitted due to space reasons.

of the preservation principle is always false, which means that the antecedent should be false in order for the principle to hold. This is so when root-preservation is satisfied. Therefore, collateral incisions over the root argument should always be avoided.

**Proposition 2** *Regarding an argument incision function “ $\sigma$ ”, if preservation is satisfied, then root-preservation is also satisfied.*

**Definition 20 (Warranting Incision Function)** *An argument incision function “ $\sigma$ ” verifying preservation is said to be a **warranting incision function**.*

In the following subsections, we will introduce the expansion and the non-warrant contraction operators in order to define the warrant-prioritized revision operator.

### 3.2. Argument Change Operators

The argument expansion can be defined in a simple manner by just adding an argument to the set of active arguments; formally:

**Definition 21 (Argument Expansion)** *An argument expansion operator “ $+^\Delta$ ” over  $T$  by a regular argument  $\mathcal{A} \in \mathbb{U}$ , namely  $T +^\Delta \mathcal{A}$ , is defined as follows:*

$$T +^\Delta \mathcal{A} = (\langle \mathbb{U}, \mathbb{A} \cup \{\mathcal{A}\}, \mathbf{R}, \sqsubseteq \rangle, \mathbf{DC})$$

Note that whenever an argument  $\mathcal{A}$  is activated, by activeness propagation every subargument in it is automatically activated. This is part of the dynamism of the theory. Moreover, the definition of the argument expansion has the inherent implications to expansions within any non-monotonic formalism: despite of the set  $\mathbb{A}$  being increased, the amount of warranted consequences could be diminished.

An argument contraction operator could be defined analogously to this expansion by incising a given argument, thus deactivating it. Nonetheless, this contraction would not be very useful towards the definition of a warrant-prioritized revision operation. Next we will define a particular kind of contraction devoted to this purpose.

This contraction operator provides warrant for an argument  $\mathcal{A} \in \mathbb{A}$  by turning every attacking line in  $\mathcal{T}_{\mathcal{A}}$  to a non-attacking line through an argument incision function  $\sigma$ .

**Definition 22 (Non-Warrant Argument Contraction)** *A non-warrant argument contraction operator “ $-^\omega$ ” of  $T$  by a regular argument  $\mathcal{A} \in \mathbb{A}$ , namely  $T -^\omega \mathcal{A}$ , is defined by means of a **warranting incision function** “ $\sigma$ ” applied over selections  $\Psi_i = \gamma(\lambda_i)$  for each  $\lambda_i \in \text{Att}_{\mathcal{A}}$  in  $\mathcal{T}_{\mathcal{A}}$ , as follows:*

$$T -^\omega \mathcal{A} = (\langle \mathbb{U}, \mathbb{A} \setminus \bigcup_i \sigma(\Psi_i), \mathbf{R}, \sqsubseteq \rangle, \mathbf{DC})$$

In contrast to the expansion, in the case of contractions the deactivation of atomic arguments by an incision involves the automatic deactivation of their superarguments.

A warrant-prioritized argument (WPA) revision operator should look for the expansion of an argument  $\mathcal{A}$  revising the warrant condition of its claim. This means that after the argument expansion “ $+^\Delta$ ” of  $\mathcal{A}$ , we should warrant its claim by effect of a non-warrant argument contraction “ $-^\omega$ ”. This operation is formally defined as follows:

**Definition 23 (Warrant-Prioritized Argument Revision)** A *warrant-prioritized argument revision operator* of  $T$  by a regular argument  $\mathcal{A} \in \mathbb{U}$ , namely  $T \times^\omega \mathcal{A}$ , is defined by means of a *warranting incision function* “ $\sigma$ ” applied over selections  $\Psi_i = \gamma(\lambda_i)$  for each  $\lambda_i \in \text{Att}_{\mathcal{A}}$  in  $T_{\mathcal{A}}$ , as follows:

$$T \times^\omega \mathcal{A} = (\langle \mathbb{U}, (\mathbb{A} \cup \{\mathcal{A}\}) \setminus \bigcup_i \sigma(\Psi_i), \mathbf{R}, \sqsubseteq \rangle, \mathbf{DC})$$

Definition 23 can be rewritten in terms of an argument expansion and a non-warrant contraction as an analogy of the Reversed Levi Identity [10]:

$$(\text{Argument Change Identity}) \quad T \times^\omega \mathcal{A} = (T +^\Delta \mathcal{A}) -^\omega \mathcal{A}$$

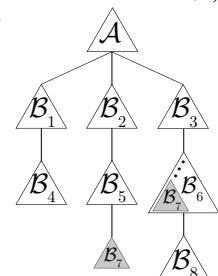
In order to warrant an argument  $\mathcal{A}$  from a theory  $T$ , a non-warranting contraction is applied considering the tree  $T_{\mathcal{A}}$ . If  $\mathcal{A} \notin \mathbb{A}$ , then  $\mathcal{A}$  is impossible to warrant since  $T_{\mathcal{A}}$  does not exist. This means that, when revising  $T$  by  $\mathcal{A}$ , first we need to expand  $T$  by  $\mathcal{A}$ , thus assuring that  $\mathcal{A}$  is active, and  $T_{\mathcal{A}}$  can be built. Therefore, the non-warranting contraction of  $T +^\Delta \mathcal{A}$  by  $\mathcal{A}$  can be performed, leading to the argument change identity.

**Theorem 1** Let  $T$  be a DAT, “ $\times^\omega$ ”, a WPA Revision Operator, and  $\mathcal{A}$ , an argument. If  $T_R = T \times^\omega \mathcal{A}$  is the DAT revised by  $\mathcal{A}$ , then  $\mathcal{A}$  is warranted from  $T_R$ .

*Proof sketch:* Let  $T_{\mathcal{A}}$  be a tree from  $(T +^\Delta \mathcal{A})$ . If  $\mathcal{A}$  is not warranted, then there is at least one attacking line ( $\lambda_i$ ). Selections over attacking lines ( $\gamma(\lambda_i)$ ) return an interference argument ( $\Psi_i$ ) responsible for that line being attacking. Incisions over the selected arguments ( $\sigma(\Psi_i)$ ) leave a subtree of  $T_{\mathcal{A}}$  containing non-attacking upper segments ( $\lambda_i^\uparrow(\Psi_i)$ ) of the former attacking lines. If uppermost collateral incisions ( $\sigma(\Psi_i)^{(\mathcal{B}_j)}$ ) occur and their upper segments ( $\lambda_j^\uparrow(\mathcal{B}_j)$ ) turn out to be attacking lines, they will be considered in concordance with the preservation principle. Finally, the tree resulting from the selection and incision process contains no attacking lines, and therefore  $\mathcal{A}$  is warranted.

#### Example 4

Let us consider a non-warrant argument contraction performed to warrant  $\mathcal{A}$ , whose tree is depicted in the figure. In this example we will select the lowest possible argument satisfying cautiousness within each attacking line in the tree. This criterion attempts to preserve the tree structure. Besides, we will use the marking function from Example 3, and assume that attacking lines are those ending with an interference argument. Regarding the line  $[\mathcal{A}, \mathcal{B}_1, \mathcal{B}_4]$ , no incision has to be performed, since it ends with a supporting argument. Line  $[\mathcal{A}, \mathcal{B}_2, \mathcal{B}_5, \mathcal{B}_7]$  is attacking and  $\mathcal{B}_2$  should be selected, since selecting  $\mathcal{B}_7$  would violate cautiousness. Finally, in the line  $[\mathcal{A}, \mathcal{B}_3, \mathcal{B}_6, \mathcal{B}_8]$ ,  $\mathcal{B}_8$  is selected and incised. The resulting argumentation lines after the contraction are  $[\mathcal{A}, \mathcal{B}_1, \mathcal{B}_4]$  and  $[\mathcal{A}, \mathcal{B}_3, \mathcal{B}_6]$ , and  $\mathcal{A}$  ends up warranted.



In the hypothetical case of a selection choosing the lowest argument regardless cautiousness, the selection in line  $[\mathcal{A}, \mathcal{B}_2, \mathcal{B}_5, \mathcal{B}_7]$  would be  $\mathcal{B}_7$ . Note that its incision would inevitably affect  $\mathcal{B}_6$ , which ends up deactivated by a collateral incision. The upper segment of  $\mathcal{B}_6$  in line  $[\mathcal{A}, \mathcal{B}_3, \mathcal{B}_6, \mathcal{B}_8]$  is  $[\mathcal{A}, \mathcal{B}_3]$ , which is an attacking line, and thereafter argument  $\mathcal{B}_3$  is selected and later on incised, because of preservation. The resulting lines after this contraction are  $[\mathcal{A}, \mathcal{B}_1, \mathcal{B}_4]$  and  $[\mathcal{A}, \mathcal{B}_2, \mathcal{B}_5]$  and  $\mathcal{A}$  is thus warranted.

The latter case shows why the property of preservation is needed: otherwise argument  $\mathcal{B}_8$  could be eligible to be incised, thus leaving the upper segment of  $\mathcal{B}_6$  (i.e.,  $[\mathcal{A}, \mathcal{B}_3]$ ) as an attacking line for a defeated root argument.

#### 4. Conclusions & Future Work

Throughout this paper, an abstract argumentation framework was proposed to be capable of dealing with knowledge dynamics. We also gave structure to arguments through the subargument relation without losing the property of being abstract. Along with this structure we defined an incomplete form of argument that can be put together with evidence in order to form an argument in the usual sense. To characterize the dynamics of the theory, we have shown how to adapt elements from the classic theory change to fit into the description of the proposed framework. The methods here introduced would be useful for an argumentation-based agent that is immersed in a changing environment.

Further analysis of the argument change operators was left as future work, including the specification of other versions of them and the definition of a set of basic postulates. Future work also includes the definition of change operators that works over the set of attack relations among arguments. This would add greater flexibility to the approach here presented, allowing for the representation of a dynamic preference criterion among arguments. Preferences could change either towards a goal or in response to a change in the “rules of the game”. A similar idea could be applied to the set of dialectical constraints. Finally, the complex composition of arguments from their subarguments, along with their multiple representations, requires further study in order to define new properties for this theory such as theory equivalence and minimal theories.

#### References

- [1] H. Prakken and G. Vreeswijk. Logical Systems for Defeasible Argumentation. In D.Gabbay, editor, *Handbook of Philosophical Logic*, 2nd ed. Kluwer Academic Pub., 2000.
- [2] C. Chesñevar, A. Maguitman, and R. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.
- [3] P. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning and Logic Programming and  $n$ -person Games. *Artificial Intelligence*, 77:321–357, 1995.
- [4] M. Falappa, G. Kern-Isbner, and G. Simari. Explanations, Belief Revision and Defeasible Reasoning. *Artificial Intelligence Journal*, 141(1-2):1–28, 2002.
- [5] M. Moguillansky, N. Rotstein, M. Falappa, and G. Simari. A Preliminary Investigation on a Revision-Based Approach to the Status of Warrant. In *Proc. of CACIC 2007*, pages 1536–1547, 2007.
- [6] I. Levi. Subjunctives, Dispositions, and Chances. *Synthese*, 34:423–455, 1977.
- [7] D. Martínez, A. García, and G. Simari. Modelling Well-Structured Argumentation Lines. In *Proc. of International Joint Conference on Artificial Intelligence IJCAI-2007 (in press)*, 2007.
- [8] M. Capobianco, C. Chesñevar, and G. Simari. An Argument-Based Framework to Model an Agent’s Beliefs in a Dynamic Environment. *Proc. of the 1st. International Workshop on Argumentation in Multiagent Systems. AAMAS 2004 Conference, New York, USA*, 3366:96–111, 2005.
- [9] C. Chesñevar and G. Simari. A Lattice-based Approach to Computing Warranted Belief in Skeptical Argumentation Frameworks. In *Proc. of the 20th Intl. Joint Conf. on Artificial Intelligence (IJCAI 2007), Hyderabad, India*, pages 280–285, January 2007.
- [10] S. Hansson. *A Textbook of Belief Dynamics: Theory Change and Database Updating*. Springer. 1999.
- [11] C. Alchourrón, P. Gärdenfors, and D. Makinson. On the Logic of Theory Change: Partial Meet Contraction and Revision Functions. *The Journal of Symbolic Logic*, 50:510–530, 1985.

# Diagramming the Argument Interchange Format

Glenn ROWE & Chris REED

*School of Computing, University of Dundee, Dundee, DD1 4HN, U.K.*

**Abstract.** We propose some standards and rules by which clear, uncluttered diagrams of arguments incorporating all the rules and attributes defined by the Argument Interchange Format (AIF) can be created. We consider issues arising from the production of general directed graph diagrams. We also define updated rules for translating between Standard, Wigmore and Toulmin diagrams that are consistent with an interpretation in terms of AIF.

## 1. The Argument Interchange Format

The Argument Interchange Format (AIF) is an international effort that aims to provide a formal framework in which all forms of monologic argument can be represented computationally. Its initial draft was presented in [4] and is being employed and further developed in the context of a range of argumentation technology projects. We will briefly outline the main concepts behind the AIF here.

The AIF is based on the idea that any monologic argument can be represented as a directed graph (digraph) called an argument network (AN). The nodes in this graph represent that various components of the argument. These nodes divide broadly into two main groups: *information nodes* (I-nodes) and *scheme nodes* (S-nodes). I-nodes contain the information used by the person making the argument to make their case. They can contain the text actually uttered or text in presumed statements found in enthymemic arguments. S-nodes describe the motivation or justification for one I-node to impinge on another. The S-node is based on the idea of an *argumentation scheme* [19]. An S-node can be one of three types: *rule of inference application* (RA-node), *preference application* (PA) or *conflict application* (CA). An RA-node is used in cases where a rule of logical inference, such as modus ponens, is used to justify the support relation provided by one or more I-nodes for another I-node. The PA-node is for expressing extra logical preferences between I-nodes and can be used for capturing (for example) values [1]. A CA-node is used when one I-node rebuts or refutes or in some (schematic) way conflicts with another I-node.

A central restriction of AIF graphs is that *no I-node may share an edge directly with another I-node*. That is, any I-node must connect to another I-node by passing through one or more S-nodes along the way. The meaning of this restriction is that a support or attack relation between I-nodes must always have a motivation or justification, and must therefore always belong to some scheme.

Note that S-node to S-node edges are allowed in the graph. This feature allows support to be provided for using a particular justification. For example, if I-node A supports I-node B, and the inferential link is provided by S-node S<sub>1</sub>, then the graph

representing this argument would show an edge from A to  $S_1$ , and an edge from  $S_1$  to B. We could then add an I-node C which gives support for the notion that it is scheme  $S_1$  (as opposed to any other scheme, say) that provides the link between A and B. In this argument, C does not directly support B, so we can't include it in the diagram by adding an edge from C to S<sub>1</sub>, since that would put C on the same footing as A as providing support for B because of scheme  $S_1$ . Rather, we wish to show that C supports  $S_1$  directly rather than B itself.

To this end, we introduce another scheme  $S_2$  which is the reason that C supports  $S_1$ , so the final diagram (Figure 1) shows an edge from C to  $S_2$  and an edge from  $S_2$  to  $S_1$ .

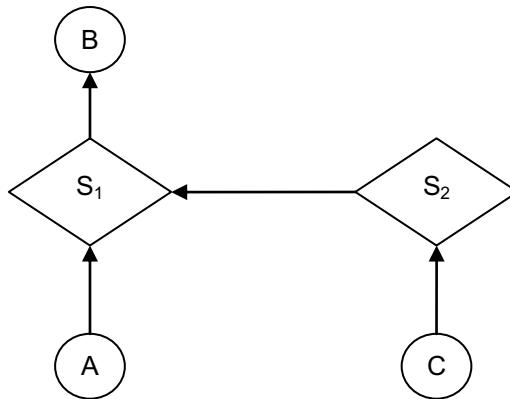


Figure 1: S-node to S-node support in AIF

Informally, then, we might say that rule-, conflict- and preference-applications can all stand as the conclusions of arguments, but that none may, on their own, stand as premises.

## 2. Software for argument diagramming

Many software packages allowing argument diagrams to be constructed interactively are currently available. A complete survey is beyond the scope of this paper, but a history of argument diagramming techniques may be found in Reed *et al.* (2007). We mention here a few existing systems.

Rationale and its predecessor Reason!Able [15], [16] are commercial packages allowing Standard diagrams to be constructed.

ArguMed [17], [18] is intended mainly for legal arguments and uses a style that is similar to Toulmin's [14] approach.

AVERs [2] produces diagrams similar to the Standard style, but also incorporates argumentation schemes.

The Carneades system [6] is a computational analysis of argument based on a philosophical model. A software package [5] based on Wigmore's [20] work has been developed for producing diagrams from the Carneades model.

Compendium [3] is a package that allows diagrams of discussions and meetings to be constructed in real-time.

This paper is concerned primarily with enhancements to the Araucaria software package allowing the full range of features defined by the AIF to be incorporated into an interactive diagramming package.

### **3. The role of Araucaria**

The AIF as originally defined is an abstract definition of an argument using the mathematical concept of a graph. The AIF standard does not propose any method by which these graphs can or should be drawn, nor does it provide a schema by which an AN can be stored or transmitted electronically. Some work has been done in devising such reifications using XML-based languages. For example, in the original AIF paper [4], an XML schema itself was proposed. Later work [8] has suggested the use of RDF schema and the Web Ontology Language OWL.

Although a representation in XML or OWL provides a clear, unambiguous representation of an argument, the production of such a representation is extremely tedious if done by hand for even a very small argument. Clearly, a software package providing a simple diagrammatic representation of an AIF graph that could then be automatically saved in XML or OWL would be of great benefit.

However, to our knowledge, no one has yet addressed the problem of defining a standard by which AIF networks can be represented as a diagram. The figures in [4] and [8] are not intended as proper diagrams of arguments; rather they are meant to illustrate the relations that are possible in AIF. As can be seen from some of these figures, attempting to represent in a diagram every feature in the AIF network leads to complex and cluttered diagrams even for simple arguments. Something cleaner and simpler is needed. We will consider some proposals in this paper.

Araucaria is a software package developed by the Argument Research Group (ARG) in the School of Computing at the University of Dundee. The most recently published version (3.1) allows an argument diagram to be built by marking up a textual document loaded from disk. Sections of text can be selected interactively and used to build an argument diagram in one of three diagram types: Standard, Toulmin and Wigmore. Araucaria 3.1 also provided a translation facility by which the user could build a diagram in any of the three supported types and then view the same diagram in the other two types.

Araucaria stores a textual representation of a diagram using Argument Markup Language (AML), which is a version of XML with a particular document type definition (DTD). The diagramming techniques used by Araucaria are implementations of the accepted diagram formats as defined for Standard, Toulmin and Wigmore diagrams. The one addition to the Standard diagram type provided by Araucaria is in the representation of schemes. Those nodes and edges forming a particular scheme can be selected and highlighted by having a coloured border or 'aura' drawn around them.

Araucaria is described in more detail in [9]. Issues related to translation between diagram types are discussed in [10] and [12]. Araucaria's use as an educational tool in critical thinking courses has been evaluated in [13].

The general acceptance of AIF as a standard for argument representation means that a new generation of diagramming tool is needed. We are currently developing Araucaria version 4 which will address this requirement. This new version faces challenges that were not present in the earlier versions of Araucaria, however, since previously we needed only to create a tool that could produce diagrams to pre-defined specifications. The formats for Standard, Toulmin and Wigmore diagrams were, for the most part, well-defined, so our job in producing a diagramming tool was precisely constrained.

The need to specify a standard for the representation of AIF diagrams, and the move to a more general argument network based on graphs, rather than the tree-based systems used for Standard, Toulmin and Wigmore diagrams pose challenges that must be met in order to produce a new version of Araucaria (version 4) which will allow general AIF diagrams to be created, edited and stored in electronic form.

#### 4. AIF Diagrams

There are two main issues that need to be faced when considering how to build AIF diagrams. The first is that an AIF diagram can be a general directed graph rather than just a tree as was the case with Standard, Toulmin and Wigmore diagrams. We therefore need a general graph-drawing interface where the nodes can be placed anywhere the user desires, rather than the more structured tree-drawing algorithm that was used in earlier versions of Araucaria.

The second issue is that of how to render the large amount of information contained in an abstract AIF network in a visible form that is as uncluttered as possible, yet still allows the user to see and edit the information easily.

The key point is the requirement that any two I-nodes must have an intervening S-node. We can get a clue as to a representation of this concept by looking at the way Araucaria 3.1 drew schemes in a Standard diagram. All nodes and edges belong to an implementation of a scheme were highlighted by having a coloured border drawn around them. The scheme itself did not appear as a distinct node in the diagram, but the user could still edit the scheme by clicking on the coloured border.

The simple situation in AIF where one or more I-nodes (the premises in an argument) support a single conclusion can be represented in this way ([Figure 2](#)). In the abstract AIF representation, all the premises lead into an S-node, and there is a single edge from the S-node to the I-node serving as the conclusion. In the diagram, the S-node is not drawn explicitly, but the premise I-nodes have edges leading directly to the I-node conclusion. All I-nodes and connecting edges are highlighted with a coloured border.

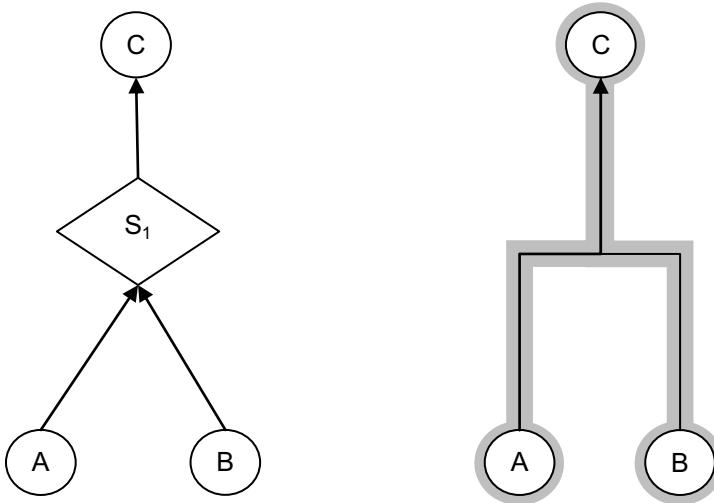


Figure 2: The AIF graph (left) shows premises A and B supporting conclusion C via scheme  $S_1$ . The equivalent diagram as visualized in Araucaria (right) does not show the scheme node explicitly, but outlines the overall argument.

Note that the necessity that any two I-nodes be connected by an S-node restricts the extent of a scheme to two layers in the graph. Any larger structures, such as A supporting B supporting C, would require two or more S-nodes to provide the links between the various layers of I-nodes.

If S-nodes are not drawn explicitly on the diagram, the complexity of the visible graph will be significantly reduced (although the underlying abstract graph will, of course, still have all the I-nodes and S-nodes present). However, the fact that an AIF graph can have S-node to S-node links leads to a problem not encountered in the earlier diagramming methods. How can we represent one S-node leading to another if the S-nodes are not drawn on the diagram?

A solution to this problem can be obtained with an inspiration from the Toulmin diagram. In a typical datum-warrant-claim Toulmin diagram, there is an edge drawn from the datum to the claim, but the warrant is connected by an edge drawn to meet the datum-claim edge at its midpoint. This reinforces the role of the warrant as a justification for making the link between the datum and the claim. To see how this can be applied to an S-node to S-node link in an AIF diagram, consider again [Figure 1](#).

In this case, the I-node C supports the S-node  $S_1$ , and does so in its turn by means of the second S-node  $S_2$ . We can represent this by drawing an edge from C to the midpoint of the edge from A to B. The scheme  $S_2$  can then be included by enclosing C and its outgoing edge in a coloured border that corresponds to this scheme. The result is shown in [Figure 3](#).

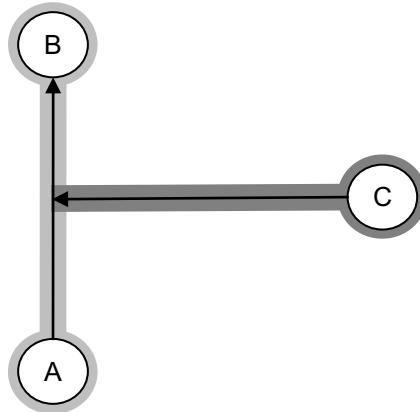


Figure 3: The representation in Araucaria of the AIF graph from Figure 1. The scheme  $S_1$  is indicated by the light grey shading; scheme  $S_2$  by the dark grey shading.

This process of supporting S-nodes can, of course, become recursive, with arbitrarily deep nesting of support, but in practice it is unlikely that this would happen over more than 2 or 3 layers. The added flexibility of nodes that the user can place as desired in the diagram mean that the spacing can be adjusted to clarify any areas of the graph that get too congested.

It is important to note here that a directed edge in the diagram does *not* necessarily imply a support relationship between the two nodes connected by the edge. Rather, a directed edge should be taken to mean a direction of inference. The inference could be support, in the sense that node A supports node B (via an intervening S-node which is not visible in the diagram) in Figure 3. However, where an edge leads from an I-node to an S-node it implies that the I-node is fulfilling a role in the scheme mediated by the S-node. There is no implication of the I-node ‘supporting’ the S-node.

Similarly, a directed edge from an S-node to an I-node (or another S-node) does not necessarily imply support. If the S-node is a conflict application (CA) node, for example, the edge represents an attack on the I-node.

The central idea behind the AIF is that all information on the nature of the inference is contained in the S-node that mediates the inference. The edges themselves indicate only the directed connections between the nodes; they do not contain any information on the nature of these connections.

## 5. Translation between Standard and Toulmin diagrams – an AIF perspective

The three diagram types supported by Araucaria 3.1 are Standard, Toulmin and Wigmore, with automatic translation provided between these types. The original paper on AIF [4] included a sample Toulmin diagram to show how it could be represented as AIF. Comparing fundamentally and theoretically different ways of conceiving of, analysing, and diagramming arguments provides evidence for the generality of representational techniques, and has been useful in the past in identifying commonalities [10][12]. We will consider here some issues that arise from the differing views of these diagram types portrayed by Araucaria and AIF.

The traditional Toulmin diagram consists of one or more data nodes that support a single claim. Each datum-claim link can be supported by a warrant, which is usually interpreted as providing justification or motivation for making the link between the datum and the claim.

From one viewpoint, it could be argued [10] that the Toulmin warrant reinforces the datum to form a linked argument supporting the claim. In this view, the Standard diagram equivalent to this Toulmin diagram is one where the datum and warrant form a pair of linked premises providing support on a equal footing for the claim. This was the view taken by Araucaria 3.1, in that linked premises in a Standard diagram were translated to a Toulmin diagram where one of the premises was (arbitrarily) singled out as the datum and the other as the warrant.

Another viewpoint [7] is that the Toulmin warrant plays the role of a scheme in the Standard diagram. Since the warrant supports the *link* between datum and claim, and not the claim directly, this seems to be a sensible position. From this standpoint, the warrant would be represented by an S-node in AIF, and this is in fact what is done in [4]. If we adopt this view, we need to consider how warrants in Toulmin and schemes in Standard translate to each other.

As we have described above, a scheme in Standard is indicated by a coloured outline around the nodes and edges that it connects, rather than by a visible node. In a Toulmin diagram, the warrant is a visible node and usually contains some text that has been selected from the document being marked up. If the Toulmin warrant is to be translated to a scheme in Standard, what properties should this scheme have? It is usual when building a Standard diagram to assign a scheme taken from a library of schemes, such as those proposed by Walton [19]. The assumption here is that there is a finite number of such schemes which can be used to categorize any type of argument. However, the Toulmin warrant usually consists of a specific span of text which relates directly to the argument being made. To use a well-worn example, the datum “Harry was born in Bermuda” supports the claim “Harry is a British subject” via the warrant “Persons born in Bermuda are British subjects”. There is no mention made of a particular scheme chosen from a standard scheme set here – the warrant is explicit in stating the reason why it supports the link between datum and claim.

One option is to require that any translation of such a diagram into a Standard diagram be done by an analyst who must decide which scheme from a standard scheme set applies to this particular argument. If we go down this route, of course, any automated translation between diagram types is impossible. Another option is to take the text of a Toulmin warrant as defining the scheme that is being used (see Figure 4). In the Bermuda example, the warrant therefore is the scheme that defines any argument in which saying someone was born in Bermuda supports the claim that that person is a British subject. Taking this approach allows an automated translation between Standard and Toulmin.

A Toulmin warrant becomes a scheme in Standard, and is drawn as a coloured outline surrounding the datum, claim and the edge connecting them. Starting with a Standard diagram, a scheme may be associated with a premise-conclusion pair by either selecting it from a scheme set such as Walton’s, or else by marking up text in the document and using this text as the definition of the scheme. The scheme is represented by a coloured outline in the Standard diagram. Translation to Toulmin displays the scheme as a warrant connecting the datum and claim. What of linked versus convergent arguments in a standard diagram? The linked/convergent distinction has long posed

theoretical challenges in the theory of argumentation [21]. Convergent arguments are independent of each other, so each can stand without any support from the others. Since each convergent argument could support the conclusion for a different reason, it is sensible to require a separate S-node for each such argument. Each convergent argument thus translates into a separate datum-warrant complex which supports the single claim.

Each component in a linked argument requires the others in order to produce support for the conclusion, so we can require that each premise in a linked argument has an edge that leads to the same S-node, and that there is a single edge from the S-node to the conclusion. Again, there is a straightforward translation to Toulmin here, where each linked premise becomes a datum node, but these data converge onto a single edge leading to the claim, with the S-node becoming the single warrant that justifies these data supporting the claim (Figure 5).

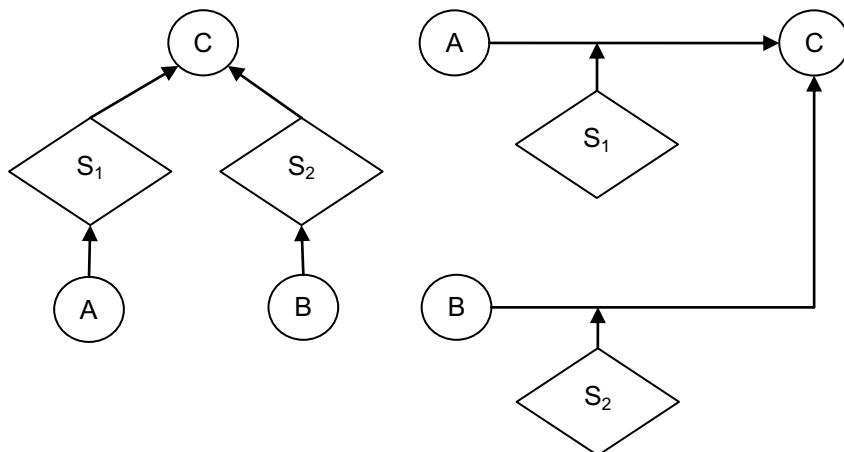


Figure 4: A convergent argument in an AIF graph (left) translates to a Toulmin diagram (right) where the schemes becomes warrants on separate datum-claim links.

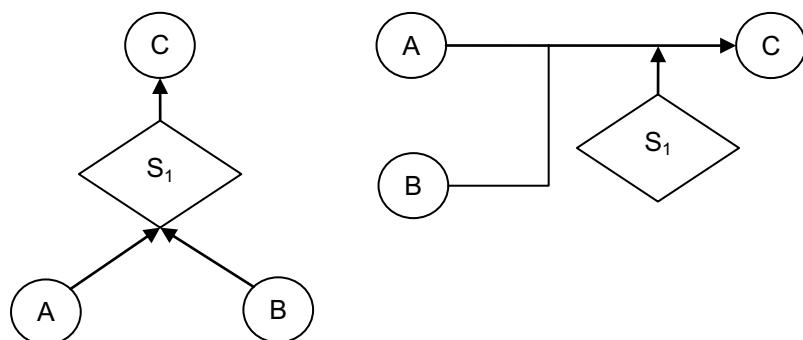


Figure 5: A linked argument in an AIF graph (left) translates to a Toulmin diagram (right) where all the data nodes are justified by a single warrant  $S_1$ .

If a Standard refutation (an I-node with label R, say) attacks directly another I-node (C, say) in the diagram, the AIF graph represents this by connecting R to a conflict application (CA) form of the S-node, which is in turn connected to C.

As has been pointed out in [8], the Toulmin rebuttal is akin to a critical question that states an exception to a given scheme. As such, the rebuttal attacks or undermines the justification (i.e. the warrant) for the datum supporting the claim, rather than attacking either datum or claim separately. In the Bermuda example, a rebuttal such as “Except if the person’s parents are American” can provide an exception to the warrant. The rebuttal can therefore be represented as an attack on the S-node represented by the warrant. The Toulmin rebuttal would translate to an I-node linked to a CA-node, which in turn links to the S-node connecting the datum and claim. See Figure 6. Of course, in some cases, a Toulmin rebuttal seems to act more like an opportunity for direct refutation of the claim. The AIF would handle such a scenario with the I-node representing the rebuttal connecting through a CA-node to the I-node representing the claim. The problem for Toulmin analyses is that they do not possess the vocabulary with which to articulate the distinction, whilst it is clear and natural to do so in the AIF and its associated diagrammatic characterisation.

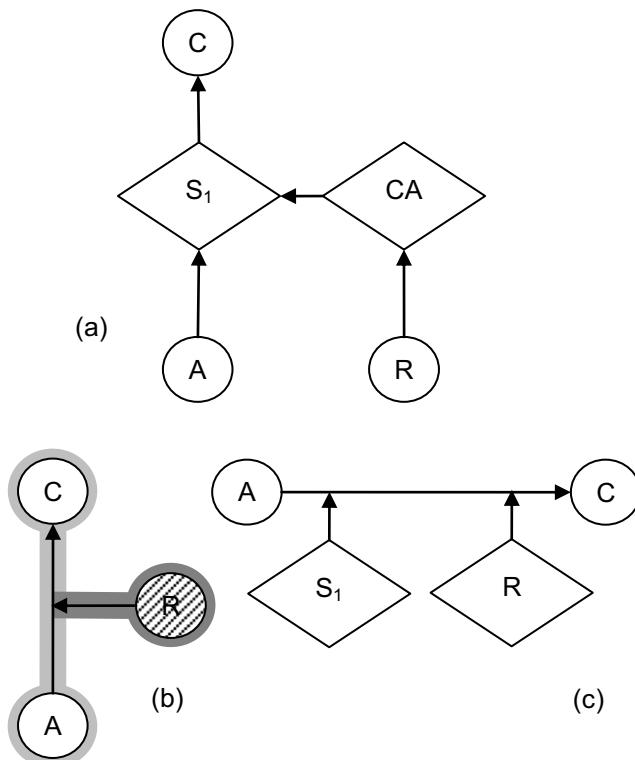


Figure 6: The Toulmin rebuttal R undermines the warrant that provides justification for the link between nodes A and C, and is thus represented in an AIF graph as shown in (a). The Standard equivalent (b) is a refutation node R (shown as shaded to distinguish it from a supporting node) which attacks the scheme by which A is claimed to support C. The dark shading surrounding R represents the CA scheme. The original Toulmin diagram is shown in (c).

## 6. Wigmore Diagrams

Wigmore diagrams [20] deal primarily with legal arguments presented in court cases. Araucaria 3.1 is the first software package to support the creation of Wigmore diagrams and to consider the issues involved in translating between Wigmore and Toulmin or Standard diagrams [12]. We consider here how such translations might fit in with the AIF concepts used above to represent Standard and Toulmin diagrams.

Nodes in a Wigmore diagram can represent evidence introduced by either the prosecution or the defense. Additionally, nodes can be divided into four main groups: testimonial, circumstantial, corroborative and explanatory. Testimonial evidence is that introduced by witness testimony, and which can be viewed as factual. Circumstantial evidence requires an inference on the part of the witness. Corroborative evidence is additional evidence provided by a witness to support their original testimonial and circumstantial evidence. Explanatory evidence provides an explanation that lessens or counters the force of the evidence introduced by the opposing side. Explanatory evidence is thus usually introduced by the side opposing the conclusion that the first three types of evidence are supporting. A simple Wigmore diagram is shown in Figure 7.

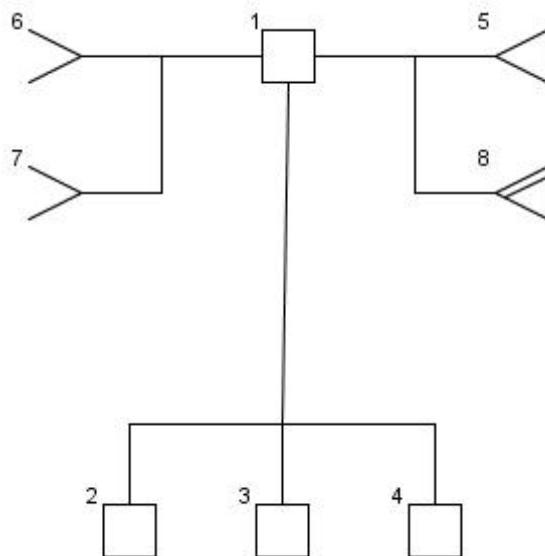


Figure 7: A simple Wigmore diagram. Node 1 is the case to be proved. Nodes 2, 3 and 4 provide testimonial evidence supporting node 1. Nodes 5 and 8 provide corroborative evidence that supports the testimonial evidence, while nodes 6 and 7 provide explanatory evidence that attempts to refute or lessen the force of the other evidence.

Wigmore diagrams also allow *forces* to be assigned to individual edges in the diagram, where each force indicates how strong or weak that particular inference is in supporting or refuting the node to which the edge leads. Issues of Wigmore forces are best dealt with by considering the evaluation of support, and as such are beyond the scope of this paper.

The edges in a Wigmore diagram indicate one node directly supporting or attacking another, with no provision for a node analogous to the Toulmin warrant or rebuttal, in which support or attack is offered to the inference between two nodes. As such, the translation between Standard and Wigmore is relatively straightforward, although we do need to make one simplifying assumption.

Although the nodes in one branch of a Wigmore diagram look superficially similar to a linked argument in a Standard diagram, Wigmore diagrams do not distinguish between convergent and linked arguments. All testimonial and circumstantial evidence is grouped together in the bottom branch, all explanatory nodes on the left, and all corroborative nodes on the right. It is certainly possible that among the nodes in one of these branches there could be one or more subsets of nodes that form separate linked arguments, while other nodes provide convergent arguments. However, there is no way of deciding this from the Wigmore diagram itself – further intervention by a human analyst would be needed.

If we wish to provide machine translation for Wigmore diagrams, the easiest solution appears to be to assume that all nodes are convergent, and thus each node has its own S-node defining how it links to the common conclusion. Since explanatory nodes argue against the central claim in the argument, it seems reasonable to assume that each explanatory node is attacking the node to which its outgoing edge leads, so we can insert a CA-node between each explanatory node and the main conclusion. All other nodes provide support for the claim, so we can use RA-nodes for these.

Clearly such an approach may miss some details of the argument, but it is not easy to see how these details could be included without further analysis of the details of the actual statements within the nodes--and further ontological extension to the upper model specified by the AIF.

## 7. Conclusion

We have defined methods for representing AIF graphs as clear, simple diagrams which can be edited using the proven software interface provided by Araucaria. Issues related to the production of general directed graphs, have been discussed, with solutions proposed. We also propose a unified way of treating features found in Standard, Toulmin and Wigmore diagrams.

## 8. Acknowledgements

We are indebted to colleagues who have given feedback on earlier drafts of this work, including, in particular, Ralph Johnson, Doug Walton, James B. Freeman and Stephen Toulmin, with whom discussions at the OSSA 2005 conference in Hamilton, Ontario have been instrumental in shaping our new account of Toulmin analyses and their link to argumentation.

## References

- [1] Bench-Capon, T., K. Atkinson and A. Chorley (2005). Persuasion and Value in Legal Argument. *Journal of Logic and Computation* 15:1075-97.

- [2] Bex, F., van den Braak, S., van Oostendorp, H., Prakken, H., Verheij, B., & Vreeswijk, G (2007). Sense-making software for crime investigation: how to combine stories and arguments? *Law, Probability and Risk*, **6**, 145-168.
- [3] Buckingham Shum, S., Selvin, A., Sierhuis, M., Conklin, J., Haley, C., & Nuseibeh, B (2006) Hypermedia Support for Argumentation-Based Rationale: 15 Years on from gIBIS and QOC, in *Rationale Management in Software Engineering* (Dutoit, A., McCall, R., Mistrik, I. and Paech, B, Eds), pp 111-132. Springer-Verlag.
- [4] Chesñevar, C. I., J. McGinnis, et al. (2006). Towards an argument interchange format, *Knowledge Engineering Review*, **21**(4): 293 – 316.
- [5] Gordon, T.F. (2007) Visualizing Carneades argument graphs. *Law, Probability and Risk*, **6**, 109-117.
- [6] Gordon, T.F., Prakken, H. & Walton, D. (2007) The Carneades model of argument and burden of proof. *Artificial Intelligence*, **171**(10-11): 875-896.
- [7] Kienpointner, M. (1992) How to Classify Arguments, in van Eemeren, F.H.,Grootendorst, R., Blair, J.A. & Willard, C.A. (eds) *Argumentation Illuminated*, pp 178 – 188, SicSat.
- [8] Rahwan, I., F. Zablit, et al. (2007). Laying the Foundations for a World Wide Argument Web, *Artificial Intelligence* **171**: 897-921.
- [9] Reed, C. and G. Rowe (2004). Araucaria: Software for Argument Analysis, Diagramming and Representation, *International Journal of AI Tools* **13**(4): 961 - 980.
- [10] Reed, C. & Rowe, G.W.A. (2005) Translating Toulmin Diagrams: Theory Neutrality in Argument Representation, *Argumentation*, **19** (3), 267-286.
- [11] Reed, C., Walton, D., & Macagno, F (2007) Argument diagramming in logic, law and artificial intelligence. *Knowledge Engineering Review*, **22**(1): 87-109.
- [12] Rowe, G.W.A. & Reed, C. (2006) Translating Wigmore Diagrams in *Proceedings of the 1st International Conference on Computational Models of Argument (COMMA 2006)*, IOS Press, 171-182.
- [13] Rowe, G., F. Macagno, et al. (2006). Araucaria as a Tool for Diagramming Arguments in Teaching and Studying Philosophy, *Teaching Philosophy* **29**(2): 111-124.
- [14] Toulmin, S. (1958) *The Uses of Argument*. Cambridge: Cambridge University Press.
- [15] van Gelder, T. (2002) A Reason!Able approach to critical thinking. *Principal Matters: The Journal for Australian Secondary School Leaders*, May 2002: 34-36.
- [16] van Gelder, T. (2007) The rationale for Rationale, *Law, Probability and Risk*, **6**, 23-42.
- [17] Verheij, B (2005) *Virtual Arguments: On the Design of Argument Assistants for Lawyers and Other Arguers*. The Hague: T.M.C. Asser Press.
- [18] Verheij, B (2007) Argumentation support software: boxes-and-arrows and beyond. *Law, Probability and Risk*, **6**: 187-208.
- [19] Walton, D. (1996). *Argumentation Schemes for Presumptive Reasoning*, Mahwah, New Jersey, Lawrence Erlbaum Associates.
- [20] Wigmore, J.H. (1931) *The Principles of Judicial Proof (2nd Edition)*; Little, Brown & Co.
- [21] Yanal, R.J. (1991) Dependent and Independent Reasons, *Informal Logic*, **13**, 137 – 144.

# Dungine: a Java Dung Reasoner

Matthew SOUTH<sup>a</sup>, Gerard VREESWIJK<sup>b</sup>, John FOX<sup>a</sup>

<sup>a</sup> Department of Engineering Science, University of Oxford, OX1 3PJ, UK

<sup>b</sup> Department of Information and Computing Sciences, Universiteit Utrecht,  
3508 TA Utrecht, The Netherlands

**Abstract.** Dungine is an open source Argumentation Engine and API implemented in the Java programming language. Dungine uses argument games to evaluate the acceptability of an argument given a constellation of arguments under grounded (sceptical) and preferred credulous semantics. Existing argumentation engines all rely on a companion logic programming language. These companion languages represent a barrier to integrating an engine into existing software as developers must translate their system into the companion logic programming language. Dungine is liberal in its notion of an Argument and this flexible approach overcomes this barrier, and makes integrating argumentation based reasoning into existing software comparatively easy. To demonstrate the usefulness of this approach, and its ease of application, we describe an integration of Dungine with Araucaria - an open source argument mapping tool.

**Keywords.** Araucaria, Argumentation Engine, Argument games, Dung, Dungine

## Introduction

Argumentation is a research topic that is applicable to many domains including legal reasoning, clinical decision support and more generally, multi-agent systems. In several of these domains, software that uses a form of argumentation within its underlying model, is becoming available. One particularly well supported paradigm in software is argument mapping where academic tools such as Araucaria<sup>1</sup> and commercial tools such as Rationale<sup>2</sup> assist users or groups of users in representing arguments concerning a particular topic. Argument mapping tools tend to focus predominantly on the structure and form of the arguments and their success depends on the willingness of their human users to accept the representation that they provide.

Representing arguments is one element of argumentation. Another important element is understanding which arguments from a set of conflicting arguments can be accepted. Dung [3] provides a formalism, agnostic of the structure of the arguments but based on an attack relation between arguments, for defining the acceptability of an argument. Within this formalism, an argument is accepted if

---

<sup>1</sup><http://araucaria.computing.dundee.ac.uk/>

<sup>2</sup><http://www.austhink.com/rationale/>

it can be successfully defended against its counterarguments. There are different ways of encoding the notion of “successfully defended against counter arguments”, which leads to different argumentation semantics. Dung defines two semantics, grounded (sceptical) semantics and preferred semantics, and defines a simple logic program for calculating the grounded acceptability of an argument. Argument games are procedural algorithms for calculating the acceptability of an argument under a given semantics. Argument games for calculating the preferred credulous acceptability of an argument are more challenging than grounded sceptical semantics and defined by Vreeswijk and Prakken [7] and Cayrolle et al [1]. Good summaries of argument game algorithms can be found in [4] and [2].

From an applications perspective, an argumentation engine that implements one or more argumentation semantics could enhance existing argument mapping systems by illustrating the accepted and defeated arguments within a particular map. Furthermore, decision support systems using argumentation could use inconsistent sources of knowledge, as long as they had some way of evaluating which arguments successfully attacked others, and autonomous agents could reason internally about their beliefs and actions.

Several implementations of the argument game algorithms for various semantics and combinations of semantics exist, including Garcia and Simari’s [5] Defeasible Logic Programming system<sup>3</sup> (DeLP), Vreeswijk’s Argumentation System<sup>4</sup> (AS), Gaertner and Toni’s CaSaPi<sup>5</sup> and the ASPIC inference engine<sup>6</sup>. All of these systems rely on a companion custom logic programming language to produce a source of arguments and a source of conflict, which limits their applicability. The Dungine engine is different to these implementations, as it defines an explicit, liberal API for consuming a source of arguments and conflicts that can be easily adapted to a wide range of argumentation sources. To use Dungine, a Java programmer must implement two interfaces, *ArgumentSource* and *DefeatSource* that provide a source of arguments and defeat relations respectively, instead of translating its source of arguments and defeats into a custom logic programming based language, as is needed for the existing solutions. This approach enables easier integration of argumentation based reasoning into existing software.

Section 1 describes the argument games that provide the theoretical basis of Dungine. Section 2 describes the Dungine API. To demonstrate the API’s ease of application, section 3 shows how Dungine can be integrated into Araucaria and discusses further potential implementations.

## 1. Argument games

Every semantics has its own argument game. Conversely, every argument game implies a certain semantics. Dungine is able to play games under the grounded semantics and the credulous version of the preferred semantics.

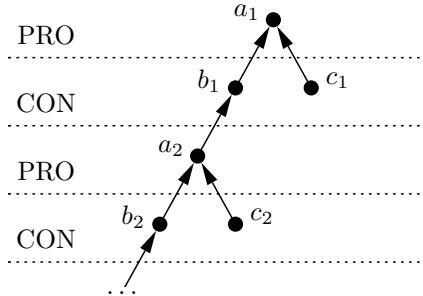
---

<sup>3</sup><http://lidia.cs.uns.edu.ar/delp>

<sup>4</sup><http://aspic.acl.icnet.uk/ArgumentationSystem/>

<sup>5</sup><http://www.doc.ic.ac.uk/~dg00/casapi.html>

<sup>6</sup>available from <http://www.argumentation.org>



**Figure 1.** Induced tree for  $a$  from example 1.2 showing the arguments that PRO and CON can play.

To explain how games are played, we first define the playing field, that is, the data-structure on the basis of which argument games are played. These are induced trees.

**Definition 1.1 (Induced tree)** Let  $G = (V, E)$  be a di-graph and let  $r \in V$ . The *tree induced by  $r$  in  $G$*  is the tree  $T$  with nodes and edges that can be reached from  $r$  by following edges backwards. To create a tree structure, nodes that re-occur through cycles are given new names but are associated with the original node.

**Example 1.2** Consider the graph  $G = \{a \rightarrow b, b \rightarrow a, c \rightarrow a\}$ . Then the tree induced by  $a$  on  $G$  is

$$T = \{a_{n+1} \rightarrow b_n, b_n \rightarrow a_n, c_n \rightarrow a_n \mid n \geq 1\}$$

Cf. Fig. 1. In this tree, every  $a_i$  corresponds to  $a$ , every  $b_i$  corresponds to  $b$ , and so forth. It is necessary to use indices to create fresh copies of  $a$ ,  $b$ , and  $c$ . The indexed nodes are supposed to represent their stem in the original di-graph.

If  $a$  is a main thesis, then argument games are played on the tree induced by  $a$ . Play alternates so PRO plays  $a$  and other arguments on odd levels, and CON plays arguments on even levels. The idea of an argument game is that, in order to demonstrate that  $a$  is undefeated, PRO must be able to show that it is able to counter all replies of CON, no matter how CON chooses its lines of attack, i.e., no matter which branch of the tree CON prefers to enter. If PRO succeeds then PRO wins, otherwise CON wins. It follows that victory is an asymmetric concept.

**Definition 1.3 (Strategy)** A *strategy for PRO* is an instruction, independent of any particular game, that describes how PRO must reply to arguments of CON.

A *winning strategy for PRO* is a strategy where PRO is guaranteed to win if PRO follows that strategy.

A similar definition for CON can be given with roles reversed.

Depending on the precise rules of the game we end up with different sorts of games, such as for example the grounded argument game, or the credulous preferred argument game.

**Definition 1.4 (Grounded argument game)** A *grounded argument game* is an argument game where PRO may not repeat or attack any of its own previous arguments in the same branch.

In a grounded argument game, CON has complete liberty, while PRO is more restricted. Still, PRO is allowed to do the following things: (i) to attack its own arguments in other branches; (ii) to advance arguments that are attacked by one of its previous arguments, possibly from another branch; (iii) to repeat one of its own arguments from another branch. Moves of type (i) and (ii) may be forbidden without changing the spectrum of outcomes, and with some gain in efficiency. A move of type (iii) is necessary for PRO to be able to reuse defenders. For example, if  $a \leftarrow b_1$  and  $a \leftarrow b_2$  and PRO is able to counter both  $b_1$  and  $b_2$  with  $c$ , then  $c$  will probably occur, as an argument of PRO, in two different branches.

**Definition 1.5 (Credulous argument game)** A *credulous argument game* is an argument game where PRO may not attack its own previous arguments, and CON may not repeat its own previous arguments.

In a credulous argument game, PRO is allowed to advance arguments that are attacked by one of its own previous arguments. This liberty can be withdrawn without changing the spectrum of outcomes, and with some gain in efficiency.

Compared to grounded argument games, credulous argument games are more pliable to PRO and less so to CON.

### 1.1. Algorithms

For every fixed argument graph and main thesis, Dungine searches a winning strategy for PRO by playing a grounded or credulous argument game where both parties are allowed to backtrack. In this way, by enacting the dialogue until CON has found a winning move, or exhausted its replies, Dungine is able to answer whether a winning strategy for PRO exists.

### 1.2. Proof representation

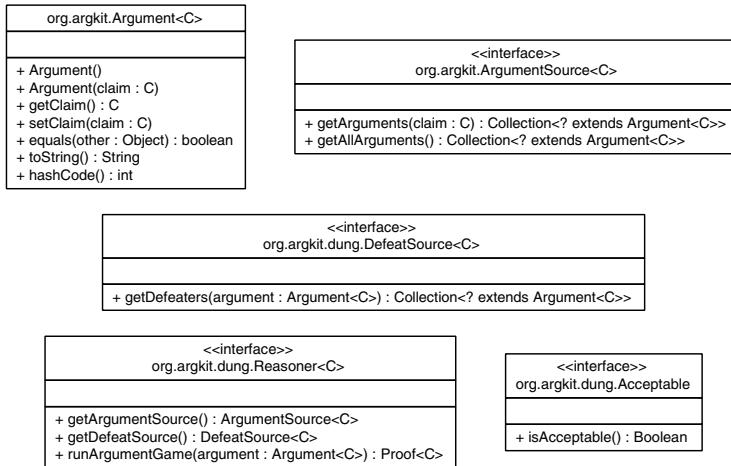
Proof representations are needed to justify the reasoning behind a particular thesis' acceptability. Dungine uses the dialogue generated by a particular argument game as the proof representation.

## 2. Implementation

Dungine is implemented in the Java programming language as part of ArgKit, an open source argumentation toolkit available from SourceForge<sup>7</sup>. The complete set of Dungine classes and their relationships is shown diagrammatically in figure 3. Dungine has a liberal interpretation of the notion of an Argument. Arguments are represented in code with a generic *Argument* class with one generic parameter, a

---

<sup>7</sup><http://argkit.sourceforge.net>



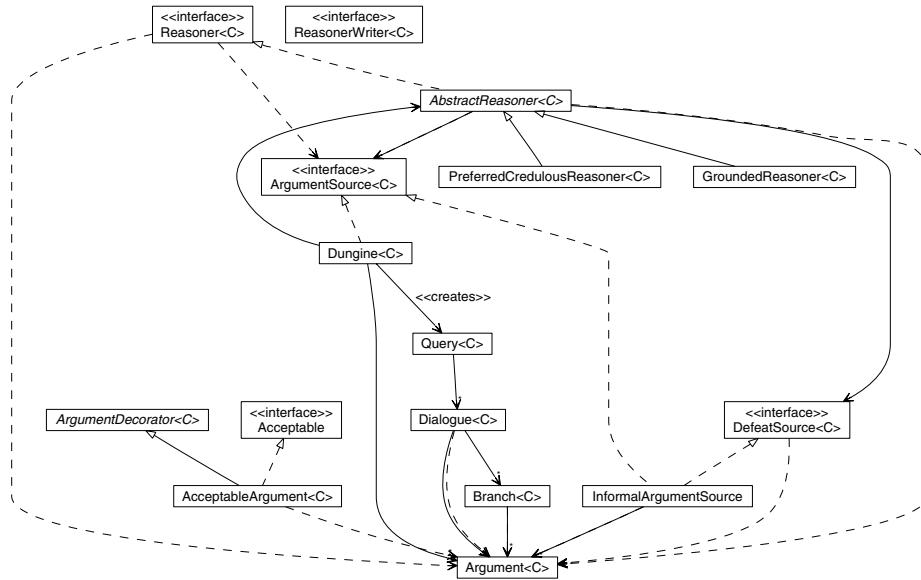
**Figure 2.** Class diagram showing the core Dungine classes and their methods.

placeholder for the class of the argument’s claim. A getter and setter for the claim is provided, along with a simple constructor and standard Java object equivalence methods (*equals* and *hashcode*) that use the argument’s claim for their notion of identity. In addition to the *Argument* class, an *ArgumentSource* interface is provided that allows a consumer to retrieve all of a provider’s arguments, or just those matching a particular claim. These classes, along with a third, *ArgumentDecorator* class represent the core of ArgKit and are contained in the *org.argkit* package. Dungine’s classes are contained in the *org.argkit.dung* package. Dungine’s key interfaces are *DefeatSource*, *Reasoner* and *Acceptable*. These core interfaces, the *Argument* class and their methods are shown in figure 2. Implementors of the *Reasoner* interface use an *ArgumentSource* and *DefeatSource* and implement a particular semantic’s argument game. An abstract, *AbstractReasoner* class is provided that manages the graph references (*ArgumentSource* and *DefeatSource*), so that implementors need only concern themselves with the argument game.

The main class is *Dungine*. This class has two use cases:

1. add an *isAcceptable* status to all arguments that it consumes,
2. provide a justification for the status evaluation of a particular argument.

Dungine consumes and publishes the *ArgumentSource* interface, and thus implements use case 1. The arguments it produces are *AcceptableArgument* objects, where each *AcceptableArgument* object references the *Argument* it consumes and decorates it with an *isAcceptable* flag. To fulfill the second use case, a *Query* object can be generated to see the justification of a particular argument’s status. For argument game based algorithms, the justification for a particular argument’s status is the induced tree from the argument game. The *Query* generation interface is based on a claim, not an argument, in order to complement the *ArgumentSource* interface. A single claim may be associated with multiple arguments, so a *Query* has a one-to-many relationship with *Dialogue* objects that provide the justification for a particular argument’s thesis.



**Figure 3.** UML class diagram showing the relationship between Dungine classes. A solid line/empty arrowhead indicates a sub-class relationship. A dashed line with an empty arrowhead indicates an "implements" relationship. A solid arrowhead indicates a class association.

### 2.1. Informal argument source

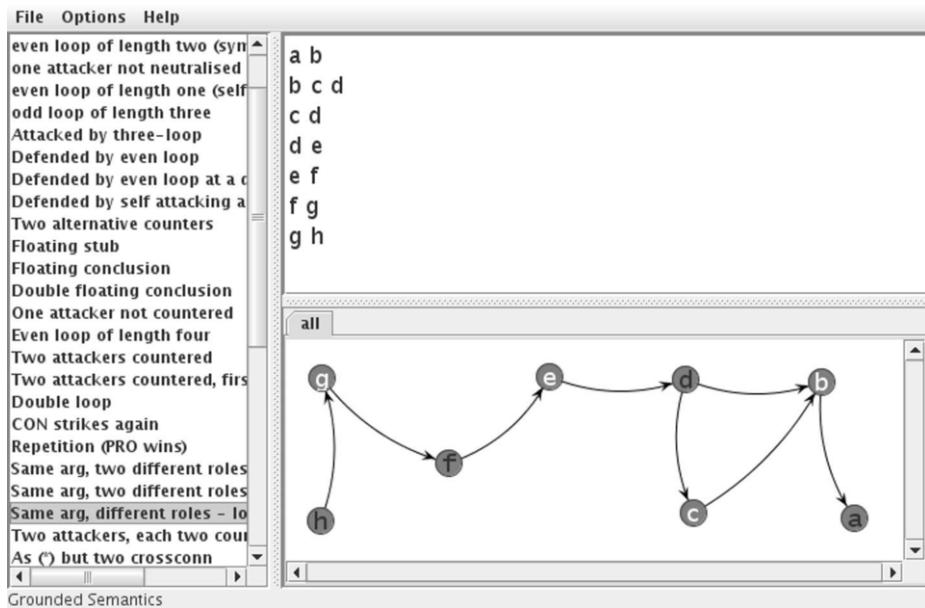
Dungine provides an informal argument source for testing purposes. The *InformalArgumentSource* class provides an immutable source of string arguments that is configured during construction with a single formatted string. Each line of the construction string starts with an argument's claim and is followed by the space delimited claims of other arguments that attack the first argument. Thus the graph used in example 1.2 earlier would be represented:

```
a b c
b a
```

Figure 4 shows a screen shot of the *TestViewer* graphical interface that is provided with Dungine to easily review the test suite that is used to validate the default reasoners. Fifty tests are defined using the informal argument source syntax. The GUI allows users to review each of the tests and the accepted status of each argument in that graph, under one of the two implemented semantics.

## 3. Applications

If the data within a Java application can be considered to have arguments that have a claim and conflict with each other, then Dungine can be used to ascertain the status of those arguments. Within the argument mapping domain, these concepts are usually explicit and thus Java argument mapping tools should gener-

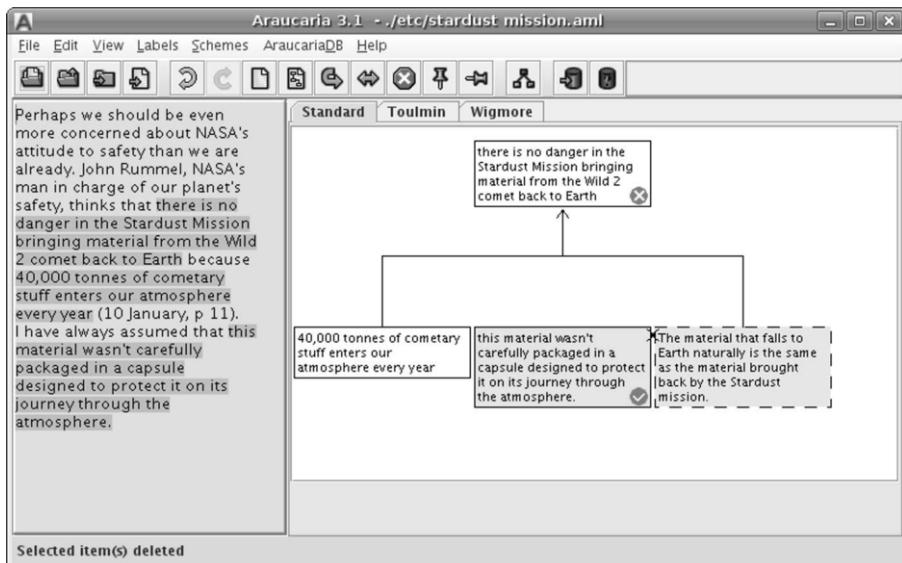


**Figure 4.** A screen shot of the GUI that provides a view of the tests used for the implemented reasoners. Tests are selected from the list in the left hand pane. Each test has an informal argument source definition (top right) and an associated graph representation (bottom right). If a graph node is accepted it is painted green (lighter - with black label) and if it is not accepted it is painted red (darker - with a white label).

ally integrate with Dungine easily. To illustrate this, the open source Araucaria<sup>8</sup> argument mapping tool has been adapted to use Dungine (see figure 5). All argument's in Araucaria are accessed via a single, top-level *Argument* object. To integrate Dungine into Araucaria, a single adapter class must be built that accepts the Araucaria *Argument* object and implements Dungine's *ArgumentSource* and *DefeatSource* interfaces. Not all of the nodes in an Araucaria graph need be considered Dungine arguments. Instead, some nodes in the Araucaria argument network can be considered to represent "main" arguments and Araucaria's notion of rebuttal can be used as the source of conflict between those main arguments. Thus the essential integration of Dungine into Araucaria can be achieved with the addition of one class of approximately fifty lines of code. In order to control the integration and visualise its results, it is necessary to introduce hooks into the Araucaria GUI to turn Dungine on and off and update the displayed arguments appropriately but this is a fairly straightforward exercise.

Thus far, Dungine has only been tested in the Argument Mapping domain. As noted in the introduction, the fields of decision support and multi-agent systems are two domains that are obvious candidates for its application. For instance Williams and Hunter [8] outline a system, OAF (Ontology Argumentation Framework) that uses ontologies to generate conflicting arguments regarding the

<sup>8</sup>see <http://araucaria.computing.dundee.ac.uk/>



**Figure 5.** A screen shot of a Dungine extended version of Araucaria. In the screen shot, Araucaria is being used to represent an argument concerning that transportation of comet material back to Earth, based on a letter to the New Scientist. Each main argument is marked with an icon in the lower right corner indicating the status of that argument. The initial argument is defeated by a counter argument provided by the letter's author.

appropriate treatment of breast cancer for a patient. Dungine could be used for the argumentation element of this system.

Generally speaking, when searching for applications for which Dungine may be a useful tool, it is often easier to identify a source of arguments than to identify and codify a source of conflict between those arguments. This does not mean that the conflict does not exist, or cannot be characterised, simply that, wherever possible it has been engineered out of software applications. It is possible that the availability of Dungine and other similar software could lead to the development of applications that were not previously considered because it allows software developers to embrace conflict within their target domains.

#### 4. Conclusion

In this paper we have introduced a new argumentation engine that uses argument games to implement Dung's grounded (sceptical) and preferred credulous semantics. To do this we have described its underlying argument game model and its Java API. A unique feature of this engine is its ease of integration. To demonstrate this, we have shown an example integration into an existing argument mapping tool, Araucaria with a single additional adapter class.

Dungine is available as part of the ArgKit project on the sourceforge website (<http://argkit.sourceforge.net>). The main direction of future work on Dungine will be towards further demonstrating applications in the domains of argument mapping, clinical decision support and multi-agent systems. An alternative di-

rection for future work is the addition of new semantics - perhaps with a new reasoner that combines Grounded and Admissible [6] semantics or a new reasoner based on the DeLP semantics [5].

## 5. Acknowledgements

The authors gratefully acknowledge the support of the EC project ASPIC (IST-FP6-002307) and Cancer Research UK for financial support during the course of this work.

## References

- [1] S. Doutre C. Cayrol and J. Mengin. On decision problems related to the preferred semantics for argumentation frameworks. *Journal of Logic and Computation*, 3(13):377–403, 2003.
- [2] P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artif. Intell.*, 170(2):114–159, 2006.
- [3] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
- [4] Leila Amgoud et al. Final review and report on formal argumentation system. Technical Report D2.6, ASPIC project, 2006. available from <http://www.argumentation.org/>.
- [5] A. Garcia and G. Simari. Defeasible logic programming: An argumentative approach. *Journal of Theory and Practice of Logic Programming*, 4:95–138, 2004.
- [6] G.A.W. Vreeswijk. An algorithm to compute minimally grounded and admissible defence sets in argument systems. In P.E. Dunne and T.J.M. Bench-Capon, editors, *Proc. of the First Int. Conference on Computational Models of Argument (COMMA06)*., pages 109–129. IOS Press, 2006.
- [7] Gerard A. W. Vreeswijk and Henry Prakken. Credulous and sceptical argument games for preferred semantics. In G. Brewka M. Ojeda-Aciego, I.P. de Guzman and L. Moniz Pereria, editors, *Proceedings of JELIA'2000, The 7th European Workshop on Logic for Artificial Intelligence*, pages 239–253, Berlin, 2000. Springer Verlag.
- [8] Matt Williams and Anthony Hunter. Harnessing ontologies for argument-based decision-making in breast cancer. In *Proceedings of the International Conference on Tools with AI (ICTAI '07)*. IEEE Press, 2007.

# Agent Dialogue as Partial Uncertain Argumentation and its Fixpoint Semantics

Takayoshi SUZUKI<sup>a</sup> and Hajime SAWAMURA<sup>b,1</sup>

<sup>a</sup>Graduate School of Science and Technology, Niigata University, Japan

<sup>b</sup>Institute of Natural Science and Technology, Niigata University, Japan

**Abstract.** Argumentation and dialogue provide important foundations for agent-oriented computing since social computing mechanism such as negotiation, cooperation, conflict resolution, etc. are to be built on them. In this paper, we consider relationship between argumentation and dialogue in a formal way. In doing so, we first describe a primitive but natural inter-agent dialogue model in which dialogue can be seen as partial argumentation that allows us to avoid excessive or unnecessary conflicts. Next, we give the semantics for such a type of dialogue, and the soundness and completeness under it.

**Keywords.** argumentation, dialogue, agreement, uncertainty, fixpoint semantics

## Introduction

Dialogue and argumentation are basic components of MAS [14], and many models for them have been studied so far (e. g., [3], [12] for argumentation, and [9], [13], [8], [2], [10] for dialogue systems). Those two are closely related to each other [17], and apparently the latter seems to be a special case of the former or one aspect of dialogue. However, it is not so clear how they are related to each other or simply how dialogue is different from argumentation. In this paper, we will consider an intrinsic relationship between them from a formal perspective.

Many argumentation models for agent-oriented computing have been developed so far [3][12][14]. Wherein, each agent is required to put forward its locutions in such a perfect form of argument that holds a logical consequence relation between a claim and reasons. Then every part of the argument are exposed to opponents and challenged by their counter-arguments (if any). This, in a sense, represents the way of argumentation in idealized situations. In our daily dialogue or argumentation, however, it happens very often that claims alone are spoken and their reasons are omitted or added later if required. This might be said to be a sort of lazy computation in the world of dialogue and argumentation.

In this paper, we describe a primitive but natural inter-agent dialogue model in which locutions such as inquiry, explanation, rebut, undercut, etc. are exchanged with each other in an imperfect or partial manner, where the explanation, for example, might not

---

<sup>1</sup>Corresponding Author: Hajime Sawamura, 8050, 2-cho, Ikarashi, Niigata, Japan. Tel/Fax: +81 25 262 6753; E-mail: sawamura@ie.niigata-u.ac.jp

have the full reasons for the question. Such a dialogue turns out to get a benefit that it allows us to avoid excessive or unnecessary conflicts, increasing chances of agreement, and hence yields productive and rich outcomes for agents concerned. We, therefore, call such a type of dialogue agreement-oriented dialogue.

As a knowledge representation language for dialogues, we use the Extended Annotated Logic Programming (EALP)[16] since it allows for a natural representation for uncertain dialogue. We think EALP is a better choice for representing uncertain knowledge than others, for example, ELP (Extended Logic Programming) [11] since it forces us to speak even uncertain locutions in an assertive tone.

We also are interested in the semantics for such a type of dialogue. We then adopt the method of the fixed point semantics initiated by Dung [5] for our dialogue system. Theoretically and practically as well, it is an interesting and significant question if dialogue systems can have a formal semantics such as the fixpoint one since it is the well-known and established one, and the most successful one for computing in general.

The paper is organized as follows. In the next section, we discuss our motivation of the paper. In Section 3, we address ourselves to formalizing agreement-oriented dialogue as partial uncertain argumentation, and illustrate a convincing dialogue. In Section 4, we formalize the fixpoint semantics for it, and give the soundness and completeness theorems that hold under the dialogical proof theory and the fixpoint semantics.

## 1. Motivation

We very often say ‘talk, rather than antagonism’ not only in our daily life but also in political, diplomatic or economic negotiations. Let us consider the following simple dialogue between two agents  $\alpha$  and  $\beta$ .

### Dialogue 1:

- $\alpha$  : The weather will be good as the sky is dyed red by the sunset. Let’s go for a drive tomorrow. [argument]
- $\beta$  : It is not a fiery sunset, but I wanna go for a drive. [counter-argument]

In terms of the standard argumentation frameworks, agent  $\beta$  is to rebut agent  $\alpha$ . So they fail to reach a full agreement on the claim ‘Let’s go for a drive tomorrow.’ What about the outcome in the next dialogue?

### Dialogue 2:

- $\alpha$  : Let’s go for a drive tomorrow.
- $\beta$  : Why? [question]
- $\alpha$  : The weather will be good tomorrow. [answer/explanation]
- $\beta$  : I also think that the weather is good tomorrow, so I accept your invitation. [agreement]

Agent  $\beta$  agrees with agent  $\alpha$ ’s invitation since for agent  $\beta$ ’s question, it could simply accept  $\alpha$ ’s reason/explanation that coincides with  $\beta$ ’s situational reason ‘I also think that the weather is good tomorrow.’, where  $\alpha$ ’s reason/explanation is partial differently from the full argument beginning with the reason ‘the sky is dyed red by the sunset’ in Dialogue 1. With this dialogue, they both could satisfy their goal for a fascinating drive.

What caused the difference of the outcome between two dialogues? The answer is that the way of presenting their locutions is different although they just begin dialoguing

with their own knowledge about desire, situation, etc. In Dialogue 1, agent  $\alpha$ 's argument gets rebut since it is fully exposed to agent  $\beta$ 's temptation to rebut. In Dialogue 2, the dialogue goes step by step without revealing every part of the argument at a time. In the second locution of agent  $\alpha$ , it speaks only the claim part, 'The weather will be good.' of agent  $\alpha$ 's original argument, 'The weather will be good as the sky is dyed red by the sunset.' (in an enthymematic representation), whose reason part was rebut by agent  $\beta$  in Dialogue 1, but is not revealed in Dialogue 2, where agent  $\alpha$  only commits to putting forward a partial argument, 'The weather will be good tomorrow.', which is shared with agent  $\beta$ . It is natural and usual that such a commonality enhances acceptability and understandability among agents.

The lesson learned from the dialogue example above is that dialogue including partial argument, enquiry, explanation, etc. can avoid excessive or unnecessary conflicts, increasing chances of agreement, and hence might yield productive and rich outcomes for agents concerned. So in this paper, we will address ourselves to formalizing agreement-oriented dialogue as partial argumentation. In doing so, we use the Extended Annotated Logic Programming (EALP) as a knowledge representation language for dialogues and the Logic of Multiple-Valued Argumentation (LMA) for uncertain argumentation since they allow for a natural representation for uncertain knowledge and argumentation[16]. For example, the unscientific and empirical but persuasive phrase 'the weather is good as the sky is dyed red by the sunset.' in Dialogue 1 above can be represented in EALP as follows: '*weather\_is\_good*:0.7 ← *sky\_is\_dyed\_red\_by\_sunset*:0.8.' We think EALP is a better choice in representing uncertain dialogue than others, for example, ELP (Extended Logic Programming) [11] since it forces us to speak even uncertain locutions in an assertive tone: '*weather\_is\_good* ← *sky\_is\_dyed\_red\_by\_sunset*.' This obviously sounds uneasy and we want to say something uncertain as it is.

## 2. Agreement-Oriented Dialogue with Uncertainty

Much work has been devoted to two-valued dialogue modes so far (e. g., [9], [13], [8], [2], [10]). However, there has been no devotion to multiple-valued dialogues despite the fact that the knowledge to be used in normal dialogues is usually uncertain. For dialogues under uncertain knowledge, we use the EALP (Extended Annotated Logic Programming), and the Logic of Multiple-Valued Argumentation (LMA) [16]. EALP is an expressive knowledge representation language that is syntactically GAP [7] extended by default negation. The many-valuedness of EALP allows agents to assert their locutions and arguments under uncertainty such as vagueness, incompleteness and inconsistency. The default negation, on the other hand, allows them to represent incomplete knowledge or beliefs.

We can exploit the diversity of truth values as complete lattices that EALP and LMA underlie, for uncertain dialogues as well. It implies the flexibility and adaptability of EALP and LMA to dialogues with various uncertainty. This aspect of EALP and LMA is most advantageous to other approaches to multiple-valued argumentation and dialogue such as [4] and [8].

## 2.1. Knowledge representation and Argument

**Definition 1 (Annotation and annotated atoms [7]).** We assume a complete lattice  $(\mathcal{T}, \leq)$  of truth values, and denote its least and greatest element by  $\perp$  and  $\top$  respectively. The least upper bound operator is denoted by  $\sqcup$ . An annotation is either an element of  $\mathcal{T}$  (constant annotation), an annotation variable on  $\mathcal{T}$ , or an annotation term. Annotation term is defined recursively as follows: an element of  $\mathcal{T}$  and annotation variable are annotation terms. In addition, if  $t_1, \dots, t_n$  are annotation terms, then  $f(t_1, \dots, t_n)$  is an annotation term. Here,  $f$  is a total continuous function of type  $\mathcal{T}^n \rightarrow \mathcal{T}$ .

If  $A$  is an atomic formula and  $\mu$  is an annotation, then  $A : \mu$  is an annotated atom. We assume an annotation function  $\neg : \mathcal{T} \rightarrow \mathcal{T}$ , and define that  $\neg(A : \mu) = A : (\neg\mu)$ .  $\neg A : \mu$  is called the epistemic explicit negation (e-explicit negation) of  $A : \mu$ .

In this paper, the e-explicit negation  $\neg A : \mu$  is embedded into an annotated atom  $A : \neg\mu$ , and implicitly handled.

**Definition 2 (Annotated literals).** Let  $A : \mu$  be an annotated atom. Then  $\sim(A : \mu)$  is the ontological explicit negation (o-explicit negation) of  $A : \mu$ . An annotated objective literal is either  $\sim A : \mu$  or  $A : \mu$ . The symbol  $\sim$  is also used to denote complementary annotated objective literals. Thus  $\sim\sim A : \mu = A : \mu$ .

If  $L$  is an annotated objective literal, then **not**  $L$  is a default negation of  $L$ , and called an annotated default literal. An annotated literal is either of the form **not**  $L$  or  $L$ .

An agent's knowledge base  $KB$  consists of the following rules.

**Definition 3 (Rules and Facts in EALP).** A rule is an expression of the form:  $H \leftarrow L_1 \& \dots \& L_n$ , where  $H$  is an annotated objective literal, and  $L_i$  ( $1 \leq i \leq n$ ) are annotated literals in which the annotation is either a constant annotation or an annotation variable. A rule with  $n = 0$  is called a fact.

For simplicity, we assume that a rule with annotation variables or objective variables represents every ground instance of it. In this assumption, we restrict ourselves to constant annotations in this paper since every annotation term in the rules can evaluate to the elements of  $\mathcal{T}$ . The head of a rule is called a *conclusion* of a rule. Annotated objective literals and annotated default literals in the body of the rule are called *antecedents* of the rule and *assumptions* of the rule respectively.

**Definition 4 (Reductant and Minimal reductant).** Suppose  $KB$  is a knowledge base in EALP, and  $C_i$  ( $1 \leq i \leq k$ ) are annotated rules in  $KB$  of the form:  $A : \rho_i \leftarrow L_1^i \& \dots \& L_{n_i}^i$ , in which  $A$  is an atom. Let  $\rho = \sqcup\{\rho_1, \dots, \rho_k\}$ . Then the following annotated rule is a reductant of  $KB$ .  $A : \rho \leftarrow L_1^1 \& \dots \& L_{n_1}^1 \& \dots \& L_1^k \& \dots \& L_{n_k}^k$ . A reductant is called a minimal reductant when there does not exist non-empty proper subset  $S \subset \{\rho_1, \dots, \rho_k\}$  such that  $\rho = \sqcup S$ .

**Definition 5 (Annotated arguments).** Let  $KB$  be a knowledge base in EALP. An annotated argument in  $KB$  is a finite sequence  $Arg = [r_1, \dots, r_n]$  of rules in  $KB$  such that for every  $i$  ( $1 \leq i \leq n$ ),

1.  $r_i$  is either a rule in  $KB$  or a minimal reductant in  $KB$ .

2. For every annotated atom  $A : \mu$  in the body of  $r_i$ , there exists a  $r_k$  ( $n \geq k > i$ ) such that  $A : \rho$  ( $\rho \geq \mu$ ) is head of  $r_k$ .
3. For every o-explicit negation  $\sim A : \mu$  in the body of  $r_i$ , there exists a  $r_k$  ( $n \geq k > i$ ) such that  $\sim A : \rho$  ( $\rho \leq \mu$ ) is head of  $r_k$ .
4. There exists no proper subsequence of  $[r_1, \dots, r_n]$  which meets from the first to the third conditions, and includes  $r_1$ .

A subargument of an argument  $Arg$  is a subsequence of  $Arg$ . The conclusions of rules in  $Arg$  are called conclusions of  $Arg$ , and the assumptions of rules in  $Arg$  are called assumptions of  $Arg$ . We write  $concl(Arg)$  for the set of conclusions and  $assm(Arg)$  for the set of assumptions of  $Arg$ . We denote the set of all arguments in  $KB$  by  $Args_{KB}$ , and define the set of all arguments in a set of EALPs,  $MAS = \{KB_1, \dots, KB_n\}$  by  $Args_{MAS} = Args_{KB_1} \cup \dots \cup Args_{KB_n}$  ( $\subseteq Args_{KB_1 \cup \dots \cup KB_n}$ ).

In this paper, we consider a dialogue whose participants consist of two agents, *proposer* and *partner* with their own knowledge in EALP and the common argumentation framework LMA.

## 2.2. Utterances

The utterances exchanged in the agent dialogue of this paper are *assertion*, *question*, and *explanation* that can be most seen in our daily dialogue. We view these as most primitive from the computational point of view as well as from the linguistic or communicational point.

**Definition 6 (Assertion)** Let  $KB$  be a knowledge base of an agent,  $Args_{KB}$  (sometimes written simply as  $Args$ ) a set of arguments in  $KB$  and  $Arg$  an argument in  $Args$ . The annotated objective literals belonging to the conclusion of the argument  $Arg$ ,  $concl(Arg)$  are called assertions in  $Arg$ . The set of all the assertions in  $Args_{KB}$  is represented by  $Ast(Args_{KB})$ .

**Definition 7 (Explanation)** Let  $A : \mu$  and  $\sim A : \mu$  be annotated objective literals, and  $A : \rho$  be an assertion in an argument  $Arg \in Args$ .

1. The explanation of  $A : \mu$  is a rule  $A : \rho \leftarrow L_1 \& \dots \& L_n$  in an argument  $Arg$ , where  $\mu \leq \rho$ .
2. The explanation of  $\sim A : \mu$  is a rule  $\sim A : \rho \leftarrow L_1 \& \dots \& L_n$  in an argument  $Arg$ , where  $\mu \geq \rho$ .

The annotated objective literals in the body of the rule  $L_1, \dots, L_n$  are called the assertions in the explanation, and the annotated default literals are called the assumptions in the explanation.

The explanations above are constrained by the conditions  $\mu \leq \rho$  and  $\mu \geq \rho$ , following the semantics of EALP and LMA (see [16] for the details of it).

**Definition 8 (Question)** Let  $L$  be an annotated objective literal. Then,  $? - L$  is a question which asks for the explanation about  $L$ .

In addition to these basic utterances, we introduce another kind of utterances that are intended to object to annotated objective literals and annotated default literals in an  $Arg$ , which are to be uttered with arguments hidden behind.

### Definition 9 (Rebut)

1. An assertion  $A:\mu_1$  rebuts an assertion  $\sim A:\mu_2$ , where  $\mu_1 \geq \mu_2$ .
2. An assertion  $\sim A:\mu_1$  rebuts an assertion  $A:\mu_2$ , where  $\mu_1 \leq \mu_2$ .

### Definition 10 (Undercut)

1. An assertion  $A:\mu_1$  undercuts **not**  $A:\mu_2$ , where  $\mu_1 \geq \mu_2$ .
2. An assertion  $\sim A:\mu_1$  undercuts **not**  $\sim A:\mu_2$ , where  $\mu_1 \leq \mu_2$ .

**Definition 11 (Attack)** Let  $L_1$  and  $L_2$  be annotated literals. Then, if  $L_1$  rebuts  $L_2$ , or  $L_1$  undercuts  $L_2$ ,  $L_1$  attacks  $L_2$ .

It is noted that these defeating relations are again followed by the semantical constraints of EALP and LMA (see [16] for the details of it).

### 2.3. Agreement-oriented dialogue and agreement

We define an agreement-oriented dialogue or simply an A-O dialogue as interacting relations between utterances described so far. First of all, we introduce a new notion of agreement for our dialogue, which reflects an idea that we can understand things for the first time when we can construct arguments on them.

**Definition 12 (Agreement)** Let  $KB_\alpha$  be a knowledge base of agent  $\alpha$ . Let  $A:\mu$  and  $\sim A:\mu$  be the assertions or the assertions in an explanation. Then,

1. agent  $\alpha$  agrees on  $A:\mu$  iff there is an  $A:\rho \in \text{Ast}(\text{Args}_{KB_\alpha})$  such that  $\mu \leq \rho$ , and
2. agent  $\alpha$  agrees on  $\sim A:\mu$  iff there is an  $\sim A:\rho \in \text{Ast}(\text{Args}_{KB_\alpha})$  such that  $\mu \geq \rho$ .

When agent  $\alpha$  agrees on  $A:\mu$  or  $\sim A:\mu$ , we say the annotated literals  $A:\mu$  or  $\sim A:\mu$  is an agreement with respect to  $A:\rho$  or  $\sim A:\rho$  of agent  $\beta$  respectively.

**Definition 13 (Agreement-oriented dialogue (A-O dialogue))** Let  $\alpha$  be a proposer agent,  $\beta$  be a partner agent,  $\text{player}_i$  be  $\alpha$  or  $\beta$ ,  $\text{utterance}_i$  be an utterance,  $\text{ast}_i$  be an assertion,  $\text{ask}_i$  be a question, and  $\text{exl}_i$  be an explanation. An agreement-oriented dialogue is a finite sequence of moves  $\text{move}_i = (\text{player}_i, \text{utterance}_i)$  ( $1 \leq i \leq n$ ) such that

1.  $\text{Player}_i = \alpha$  iff  $i$  is an odd number, and  $\text{Player}_i = \beta$  iff  $i$  is even number;
2. if  $\text{Player}_i = \text{Player}_j$  ( $i \neq j$ ),  $\text{utterance}_i \neq \text{utterance}_j$ ,
3. if  $\text{move}_i = (\text{Player}_i, \text{ast}_i)$  and  $\text{Player}_{i+1}$  doesn't agree on  $\text{ast}_i$  (i. e.,  $\text{ast}_i \notin \text{Ast}(KB_{\text{player}_{i+1}})$ ),  $\text{move}_{i+1} = (\text{Player}_{i+1}, \text{ask}_{i+1})$ ,
4. the dialogue starts with  $\text{move}_1 = (\alpha, \text{ast}_1)$ , and then  $\text{move}_2 = (\beta, \text{ast}_2)$ , where  $\text{ast}_2$  rebuts  $\text{ast}_1$ , or  $\text{move}_2 = (\beta, \text{ask}_2)$ ,
5. if  $\text{move}_i = (\text{player}_i, \text{ast}_i)$ , where  $\text{ast}_i$  rebuts an annotated objective literal asserted before, then  $\text{move}_{i+1} = (\text{player}_{i+1}, \text{ask}_{i+1})$ ,
6. if  $\text{move}_i = (\text{player}_i, \text{ast}_i)$ , where  $\text{ast}_i$  undercuts an annotated default literal asserted before, then either  $\text{move}_{i+1} = (\text{player}_{i+1}, \text{ast}_{i+1})$ , where  $\text{ast}_{i+1}$  rebuts  $\text{ast}_i$  or  $\text{move}_{i+1} = (\text{player}_{i+1}, \text{ask}_{i+1})$ ,

7. if  $move_i = (player_i, ask_i)$ , then  $move_{i+1} = (player_{i+1}, exl_{i+1})$ , and
8. if  $move_i = (player_i, exl_i)$ , where  $exl_i$  is not a fact, then either  $move_{i+1} = (player_{i+1}, ast_{i+1})$ , where  $ast_{i+1}$  attacks the antecedent of  $exl_i$ , or  $move_{i+1} = (player_{i+1}, ask_{i+1})$ .

The condition 3 of Definition 13 above means that if an assertion put forward is not included in the assertions in the hearer's knowledge base, questions are always uttered (such a curiosity may be a kind of agent attitude or personality). It should be noted that no moves are possible if  $exl_i$  in the condition 8 is a fact, as can be seen in Fig. 1 below.

**Definition 14 (Agreement-oriented dialogue tree)** *An agreement-oriented dialogue tree (simply an A-O dialogue tree.) is a tree of dialogue moves such that*

1.  $move_1$  is the root, and
2. the children of  $move_i$  are all  $move_{i+1}$ s that satisfy Definition 13, where
  - a child of  $move_i$  is  $move_{i+1} = (player_{i+1}, agreement\ ast_i)$  if  $player_{i+1}$  has agreed on the assertion  $ast_i$  or assertions in the explanation for  $ast_i$  by  $player_i$ .
  - a child of  $move_i$  is  $move_{i+1} = (player_{i+1}, disagreement\ ast_i)$  if  $move_i = (player_i, exl_i)$ , where  $exl_i = ast_i \Leftarrow$ , i. e., the explanation  $exl_i$  of  $player_i$  is a fact.

Finally, we introduce definitions to specify the outcomes of A-O dialogues based on the A-O dialogue tree. The following notion is a counterpart of the notion of *dialectical justification* of arguments [11].

**Definition 15 (Dialogical agreement)** *Let  $Ast$  be an assertion put forward by a proposer.  $Ast$  is dialogically agreed if there exists an A-O dialogue tree whose root is  $Ast$  and every leaf of the A-O dialogue tree is a partner's utterance of the form  $(player, agreement\ ast)$ .*

This definition yields a total or strong agreement in which participant agents have to reach an agreement for each possible dialogue flow. We also have a weaker one that could capture daily dialogic phenomena.

**Definition 16 (Partial agreement)** *Let  $Ast$  be an assertion put forward by a proposer.  $Ast$  is partially or weakly agreed if there exists an A-O dialogue tree such that the root is  $Ast$  and at least one leaf is not a partner's utterance of the form  $(player, agreement\ ast)$ .*

### 3. A-O Dialogue Example (on Nuclear Power Plant)

Two agents  $\alpha$  (a proposer) and  $\beta$  (a partner) have the knowledge bases  $KB_\alpha$  and  $KB_\beta$  about the nuclear power plants (NPP) respectively. They begin an  $A - O$  dialogue about whether a nuclear power plant is dangerous or not. We use the lattice of the unit interval of reals,  $\Re = < [0, 1], \leq >$  as the truth values for the example.

$$KB_\alpha = \left( \begin{array}{l} r_\alpha 1 : \text{depressed}(Japan):1.0 \leftarrow \\ r_\alpha 2 : \text{shortage of budget}(Japan):0.8 \leftarrow \text{depressed}(Japan):1.0 \\ r_\alpha 3 : \text{superannuated equipment}(NPP):1.0 \leftarrow \\ \quad \text{shortage of budget}(Japan):0.8 \\ r_\alpha 4 : \text{radiation leak}(NPP):0.5 \leftarrow \\ \quad \text{superannuated equipment}(NPP):0.8 \\ r_\alpha 5 : \text{pollute environment}(NPP):1.0 \leftarrow \text{radiation leak}(NPP):0.5 \\ r_\alpha 6 : \text{dangerous}(NPP):1.0 \leftarrow \text{pollute environment}(NPP):0.8 \\ r_\alpha 7 : \text{save(electricpower)}:0.5 \leftarrow \\ r_\alpha 8 : \sim \text{insufficient(electric power)}:1.0 \leftarrow \text{save(electric power)}:0.5 \\ r_\alpha 9 : \text{oppose}(NPP):1.0 \leftarrow \text{dangerous}(NPP):1.0 \\ \quad \& \sim \text{insufficient(electric power)}:0.8 \\ r_\alpha 10 : \text{many earthquakes}(Japan):1.0 \leftarrow \\ r_\alpha 11 : \text{cause accident}(NPP):1.0 \leftarrow \text{many earthquakes}(Japan):0.8 \\ r_\alpha 12 : \text{produce radioactive waste}(NPP):1.0 \leftarrow \\ r_\alpha 13 : \sim \text{environment-friendly}(NPP):0.8 \leftarrow \\ \quad \text{produce radioactive waste}(NPP):1.0 \end{array} \right)$$

$$KB_\beta = \left( \begin{array}{l} r_\beta 1 : \sim \text{dangerous}(NPP):1.0 \leftarrow \text{not cause accident}(NPP):0.5 \\ r_\beta 2 : \text{many earthquakes}(Japan):1.0 \leftarrow \\ r_\beta 3 : \sim \text{pollute environment}(NPP):0.5 \leftarrow \\ \quad \text{not } \sim \text{environment-friendly}(NPP):0.8 \\ r_\beta 4 : \text{produce radioactive waste}(NPP):1.0 \leftarrow \\ r_\beta 5 : \text{artificial mistake}(NPP):0.5 \leftarrow \\ r_\beta 6 : \text{radiation leak}(NPP):1.0 \leftarrow \text{artificial mistake}(NPP):0.5 \\ r_\beta 7 : \text{save(electricpower)}:0.8 \leftarrow \\ r_\beta 8 : \sim \text{superannuated equipment}(NPP):0.5 \leftarrow \end{array} \right)$$

We suppose that agent  $\alpha$  starts doing the first assertion. Then the dialogue progress for the NPP dialogue is shown in a dialogue tree in Fig. 1, where every leaf of the dialogue tree ends with an agreement of  $\beta$ . The first assertion brought up by Agent  $\alpha$ , [ $\text{be dangerous}(NPP):1.0$ ], therefore, is dialogically agreed by  $\beta$ . Put it differently, it can be said that  $\alpha$  could persuade  $\beta$ .

Let us scrutinize the following rules which did not appear in the dialogue process:

$$KB'_\alpha = \left( \begin{array}{l} r_\alpha 1 : \text{depressed}(Japan):1.0 \leftarrow \\ r_\alpha 2 : \text{shortage of budget}(Japan):0.8 \leftarrow \text{depressed}(Japan):1.0 \\ r_\alpha 3 : \text{superannuated equipment}(NPP):1.0 \leftarrow \\ \quad \text{shortage of budget}(Japan):0.8 \\ r_\alpha 4 : \text{radiation leak}(NPP):0.5 \leftarrow \\ \quad \text{superannuated equipment}(NPP):0.8 \end{array} \right)$$

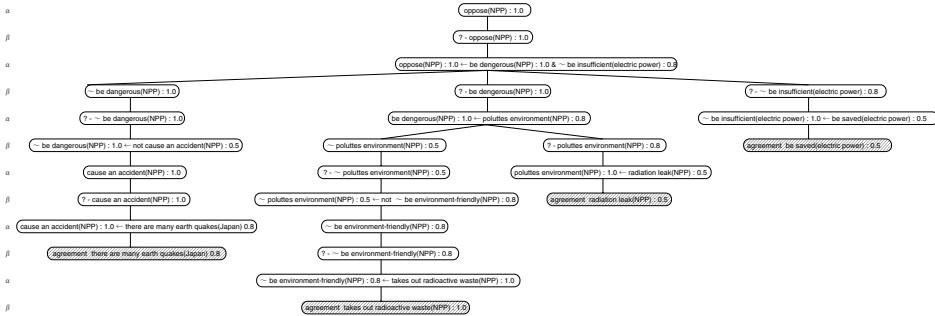


Figure 1. The A-O dialogue tree for the NPP dialogue

$$KB'_\beta = \left( \begin{array}{l} r_\beta 5 : \text{artificial mistake}(NPP) : 0.5 \leftarrow \\ r_\beta 6 : \text{produce radioactive waste}(NPP) : 1.0 \leftarrow \\ \quad \text{artificial mistake}(NPP) : 0.5 \\ r_\beta 8 : \sim \text{superannuated equipment}(NPP) : 0.5 \leftarrow \end{array} \right)$$

Since  $\alpha$ 's assertion [radiation leak(NPP) : 0.5] of  $r_\alpha 4$  can be agreed by  $\beta$ , the antecedent for it has not had a chance to be exposed to  $\beta$  in the dialogue. With a complete argumentation, they could not reach an agreement since the assertion in explanation of  $r_\alpha 4$  [superannuated equipment(NPP) : 1.0] and the assertion of  $r_\beta 8$  [ $\sim$  superannuated equipment(NPP) : 0.5] are in the rebut relation. In the argumentation which attacks recklessly by the rebut or undercut, even if there is a portion which can agree with the opinion of the partner, it is disregarded and the argumentation can not reach an agreement after all in many cases. As we have attempted in this paper, the utterance is not a full argument but an assertion that is a partial argument in the sense that its full argument is not exposed to the other party. As the result, it led to such an agreement-oriented dialogue that the dialogue can go to a success in the case that the conclusion part is agreed but a portion in the antecedent of a rule conflicts with each other.

The example above illustrates our first conclusion that *dialogue is partial argumentation* as suggested in the paper title. And this is just the way how dialogue including partial argument, enquiry, explanation, etc. can avoid excessive or unnecessary conflicts, increasing chances of agreement, and hence may yield productive and rich outcomes for agents concerned. In the next section, we go about the question of how a formal semantics can be given for such an agreement-oriented dialogue.

#### 4. Fixpoint Semantics for the Agreement-Oriented Dialogue

We will consider the so-called fixpoint semantics for the agreement-oriented dialogue since it is the well-established and the most successful semantics for argumentation frameworks. It originates from Dung's influential work [5], and further has been applied to modified or extended argumentation frameworks so far by many authors (e. g., [11], [1], [16]).

However, there seems to be not much work on the formal semantics for agent dialogues although they have been proposed in one way or another for the past few years

(e. g., [9], [13], [8], [2], [10]). In this section, we give a fixpoint semantics for our agreement-oriented dialogue in a similar manner to those of argumentation frameworks [3][12].

First and foremost, we introduce the notion of ‘agreeable’ that is a counterpart of ‘acceptable’ in argumentation frameworks [3][12], and plays an essential role even in the development of dialogue semantics in what follows.

**Definition 17 (Agreeable)** Let  $KB$  be a knowledge base,  $Asts(KB)$  be a set of the assertions in a knowledge base  $KB$  (simply written as  $Asts$ ),  $S \subseteq Asts$ , and  $Ast \in Asts$ . Then,  $Ast$  is agreeable with respect to  $S$  if and only if

1. if an assertion  $\neg Ast \in Asts$  attacks  $Ast$ , then at least one of the literals in the antecedent of the explanation of  $\neg Ast$  is attacked by the assertions in  $S$  [case of explanation for question  $? - \neg Ast$ ].
2. if the assertion  $Ast$  is asked by a question  $? - Ast$ , then all the assertions in the explanation of  $Ast$  are agreeable with respect to  $S$  and the assumptions in the explanation of  $Ast$  are not undercut by an assertion in  $S$  [case of question  $? - Ast$ ], and
3. if the assertion  $Ast$  is in  $S$ , then it is agreeable with respect to  $S$ .

This definition captures the notion of ‘agreeableness’ that corresponds to the notion of ‘acceptability’ in argumentation. We have incorporated it in the context of dialogue. Based on agreeableness, we define ‘agreed assertions’ as a counterpart of ‘justified argument’ by a fixpoint construction. See [15] for the proofs of the propositions and theorems below.

**Definition 18 (Characteristic function)** Let  $S$  be a subset of  $Asts$ .  $F_{Asts}$  is a characteristic function such that

- $F_{Asts} : Pow(Asts) \rightarrow Pow(Asts)$
- $F_{Asts}(S) = \{A \in Ast(Arg_{KB_\alpha}) \mid A \text{ is agreeable with respect to } S\}$ .

**Proposition 1**  $F_{Asts}$  is monotonic with respect to set inclusion.

Since  $F_{Asts}$  is monotonic, it has a least fixpoint  $\text{lfp}(F_{Asts})$ , and can be constructed by the iterative method below. The following is a counterpart of the notion of ‘justified’ in argumentation frameworks [11].

**Definition 19 (Agreed)**  $A \in Ast(Arg_{KB_\alpha})$  is agreed iff the assertion  $A$  is in the least fixed-point of  $F_{Asts}$  (from now on, it is represented as  $JustAsts$ .)

**Definition 20 (Finitary [5])**  $Asts$  is finitary iff each assertion in  $Asts$  is attacked by at most a finite number of assertions in  $Asts$ .

With these preliminaries, we can state the following proposition for our A-O dialogue, which allows for calculating the fixpoint in the iterative method.

**Proposition 2** Let  $KB_\alpha$  and  $KB_\beta$  be the knowledge bases of agents  $\alpha$  and  $\beta$  respectively. We define a sequence of subsets of agent  $\alpha$ ’s assertions  $Asts(KB_\alpha)$  as follows:

- $F^0 = \{A \in \text{Ast}(\text{Arg}_{KB_\alpha}) \mid A \text{ is an agreement wrt. an assertion } B \in \text{Ast}(\text{Arg}_{KB_\beta})\}$
- $F^{i+1} = F_{\text{Ast}s}(F^i)$

*Then, if  $\text{Ast}s$  is finitary,  $\cup_{i=0}^{\infty}(F^i) = \text{JustAst}s$ .*

$F^0$  in Proposition 2 is unique and idiosyncratic in A-O dialogues, compared with its counterpart in argumentation frameworks [5][11], where it starts with  $F_0 = \phi$  since justified arguments should be accrued from self-helped arguments. Our definition is based on the fact that in an A-O dialogue, when an assertion is dialogically agreed, the A-O dialogue tree must end with agreements of the partner at all the leaves of it (see Fig. 1).

For the agreement-oriented dialogue, we have the desirable properties: soundness and completeness, as in standard argumentation frameworks [5][11].

**Theorem 1 (Soundness)** *If an assertion is dialogically agreed, then it is agreed.*

**Theorem 2 (Completeness)** *If an assertion is agreed, then it is dialogically agreed.*

## 5. Related work

There have been some works on uncertain argumentation frameworks under uncertain knowledge base, such as possibilistic argumentation framework [4], probabilistic argumentation [6], etc. However, there seems to be no work on uncertain dialogue models and their formal semantics. In these circumstances, the present paper is considered a challenging attempt towards uncertain dialogue and formal dialogue semantics. Nevertheless, the dialogue system of this paper is most primitive, so that the locutions are basically ones included in most of (argument-based) dialogue systems proposed so far (e. g., [9], [13], [8], [2], [10]). It can be roughly similar to persuasion dialogue in the typology of human dialogues by Walton et. al. [17].

Risk Agoras is a dialogue system for scientific reasoning [8]. It includes a locution *show\_arg()* that can require an argument as a partial one for a *query()*. This aspect of the dialogue is similar to the idea of this paper: dialogue as partial argument. In order to deal with uncertain dialogues in natural language, Risk Agoras also incorporates dictionaries for modalities for claims, grounds, consequences and rules of inference whether quantitative or qualitative. Risk Agoras, however, has no formal semantics for its dialogue. Prakken's dialogue framework [10] is most closely related to the present paper although the former is much more versatile and flexible than ours in many ways. We would say that Prakken introduced many notions involved in dialogue systems and examined their properties. Two main attentions that he has not been paid to are those issues on uncertainty in dialogue and the fixpoint semantics. In this paper, we concentrated our attention on the question if there can be the fixpoint semantics for uncertain dialogue systems in a similar way to that for argumentation.

## 6. Conclusion and Future work

We have presented a new type of dialogue system called agreement-oriented dialogue and its fixpoint semantics as the formal semantics for dialogues.

We summarize the contributions of the paper as follows:

- We characterized uncertain dialogue as partial argumentation, which we think is a natural aspect of human dialogues. We revealed a difference between dialogue and argumentation by exemplifying that there is an issue which is not justified with argumentation only but agreed in the context of a dialogue.
- We showed that the agreement-oriented dialogue system could have the fixpoint semantics, similarly to those for argumentation frameworks. This suggests a possibility of formal or mathematical semantics even for actual dialogues or conversations in our daily life.
- We gave the soundness and completeness theorems for the agreement-oriented dialogue system.

We will further develop the motto ‘Dialogue as partial uncertain argumentation’ to more complicated uncertain dialogues on uncertain issues.

## References

- [1] L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–215, 2002.
- [2] L. Amgoud, N. Maudet, and S. Parsons. Modeling dialogues using argumentation. In *ICMAS*, pages 31–38, 2000.
- [3] C. I. Chesñevar, G. Maguitman, and R. P. Loui. Logical models of argument. *ACM Computing Surveys*, 32:337–383, 2000.
- [4] C. I. Chesñevar, G. Simari, T. Alsinet, and L. Godo. A Logic Programming Framework for Possibilistic Argumentation with Vague Knowledge. In *Proc. of the Intl. Conference on Uncertainty in Artificial Intelligence (UAI2004)*, pages 76–84, 2004.
- [5] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–358, 1995.
- [6] R. Haenni, J. Kohlas, and N. Lehmann. Probabilistic argumentation systems. In *In J. Kohlas and S. Moral, editors, Handbook of Defeasible Reasoning and Uncertainty Management Systems, Vol. 5: Algorithms for Uncertainty and Defeasible Reasoning*, pages 221–288. Kluwer, 2000.
- [7] M. Kifer and V. S. Subrahmanian. Theory of generalized annotated logic programming and its applications. *Journal of Logic Programming*, 12(3,4):335–367, 1992.
- [8] P. McBurney and S. Parsons. Risk agoras: Dialectical argumentation for scientific reasoning. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI-2000)*, pages 371–379, 2000.
- [9] S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *J. of Logic and Computation*, 13(3):347–376, 2003.
- [10] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *J. of Logic and Computation*, 15(6):1009–1040, 2005.
- [11] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *J. of Applied Non-Classical Logics*, 7:25–75, 1997.
- [12] H. Prakken and G. Vreeswijk. Logical systems for defeasible argumentation. In *In D. Gabbay and F. Guenther, editors, Handbook of Philosophical Logic*, pages 219–318. Kluwer, 2002.
- [13] C. Reed. Dialogue frames in agent communication. In *Proceedings of the 3rd International Conference on Multi Agent Systems (ICMAS98)*, pages 246–253, Paris , France, 1998.
- [14] C. Reed and T. J. Norman, editors. *Argumentation Machines*. Kluwer Academic Publishers, 2004.
- [15] T. Suzuki. Agent dialogue as partial argumentation and its fixpoint semantics. Master Thesis, Niigata University, 2007. (in Japanese).
- [16] T. Takahashi and H. Sawamura. A logic of multiple-valued argumentation. In *Proceedings of the third international joint conference on Autono mous Agents and Multi Agent Systems (AAMAS'2004)*, pages 800–807. ACM, 2004.
- [17] D. Walton. *The New Dialectic: Conversational Contexts of Argument*. Univ. of Toronto Press, 1998.

# A Distributed Argumentation Framework using Defeasible Logic Programming

Matthias Thimm <sup>a</sup> Gabriele Kern-Isberner <sup>a</sup>

<sup>a</sup> *Information Engineering Group, Faculty of Computer Science  
Technische Universität Dortmund, Germany*

**Abstract.** *Defeasible Logic Programming* (DeLP) by García and Simari is an approach to realise non-monotonic reasoning via dialectical argumentation. We extend their approach by explicitly supporting distributed entities in the argumentation process on a structural basis. This makes the modelling of distributed argumentation systems like a jury court by using DeLP techniques possible. In this framework possibly many different agents with different opinions argue with each other on a given logical subject. We compare our framework with general DeLP and present the results.

**Keywords.** Logic Programming, Argumentation, Defeasible Argumentation, Distributed Argumentation, Multi Agent Systems.

## 1. Introduction

Mimicking *commonsense-reasoning* using non-monotonic logics is one of the main topics in AI. *Defeasible Logic Programming* (DeLP) [5] is an approach to realise non-monotonic reasoning via dialectical argumentation by relating arguments and counterarguments for a given logical query. A dialectical process that considers all arguments and counterarguments for the query is used in order to decide whether the query is believed by the agent or not. So in DeLP argumentation is treated as an internal deliberation mechanism of one agent to determine the set of pieces of information which are most believed.

But in general the term *argumentation* is much more abstract. It can also be the exchange of arguments and counterarguments between several agents in a multi agent environment where every agent tries to convince other agents of a specific opinion. Consider a jury court, where every juror has a personal opinion about the guilt or innocence of an accused person. The jurors give arguments for one or the other and attack other jurors' arguments with counterarguments. In the end one argument may prevail and its conclusion is given to the initiator of the query, e. g. the judge.

There are many approaches to realize negotiation in multi agent systems. Whereas in [1,7] and especially in [2], the focus is on using argumentation for persuasion, in this paper we use argumentation to reach a common conclusion of a group of agents. Considering the jury court it is reasonable to assume that there

are jurors who are less competent in jurisdiction than others. However it is the main goal to reach an agreement regarding the given case rather than unifying the jurors beliefs.

This paper proposes and discusses an approach for a distributed system which provides the capability of argumentation using the notions of DeLP. In this system agents exchange arguments and counterarguments in order to answer queries given from outside the system. The framework establishes a border between its interior and exterior as from outside the system it is seen as a general reasoning engine. Internally this reasoning is accomplished by defeasible argumentation where every agent tries to support or defeat the given query by generating arguments for or against it and by generating counterarguments against other agents' arguments. In the end the most plausible argument prevails and its conclusion is the answer to the original query.

The rest of this paper is structured as follows: in Section 2 we give a brief overview on DeLP. In Section 3 a framework for modelling distributed argumentation using DeLP is proposed. Section 4 compares the framework with general DeLP and in Section 5 we conclude.

## 2. Defeasible Argumentation

Defeasible Logic Programming (DeLP) [5] is a logic programming language which is capable of modelling defeasible knowledge. With the use of a defeasible argumentation process it is possible to derive conclusive knowledge.

The basic elements of DeLP are facts and rules. The set of rules is divided into strict rules, i. e. rules which derive certain knowledge, and defeasible rules, i. e. rules which derive uncertain or defeasible knowledge. We use a first-order language without function symbols except constants, so let  $\mathcal{L}$  be a set of literals, where a literal  $h$  is atom  $A$  or a negated atom  $\neg A$ , where the symbol  $\neg$  represents the strong logic negation. Overlining will be used to denote the complement of a literal with respect to strong negation, i. e. it is  $\overline{p} = \neg p$  and  $\overline{\neg p} = p$  for a ground atom  $p$ .

**Definition 1** (Fact, strict rule, defeasible rule). A *fact* is a literal  $h \in \mathcal{L}$ . A *strict rule* is an ordered pair  $h \leftarrow B$ , where  $h \in \mathcal{L}$  and  $B \subseteq \mathcal{L}$ . A *defeasible rule* is an ordered pair  $h \prec B$ , where  $h \in \mathcal{L}$  and  $B \subseteq \mathcal{L}$ .

A defeasible rule is used to describe uncertain knowledge as in “birds fly”. We use the functions *body*/1 and *head*/1 to refer to the head resp. body of a defeasible or strict rule.

**Definition 2** (Defeasible Logic Program). A *Defeasible Logic Program*  $\mathcal{P} = (\Pi, \Delta)$ , abbreviated *de.l.p.*, consists of a (possibly infinite) set  $\Pi$  of facts and strict rules and of a (possibly infinite) set  $\Delta$  of defeasible rules.

**Example 1** ([5], example 2.1). Let  $\mathcal{P} = (\Pi, \Delta)$  be given by

$$\Pi = \left\{ \begin{array}{ll} \text{chicken}(tina) & \text{scared}(tina) \\ \text{penguin}(tweety) & (\text{bird}(X) \leftarrow \text{chicken}(X)) \\ \text{bird}(X) \leftarrow \text{penguin}(X) & (\neg \text{flies}(X) \leftarrow \text{penguin}(X)) \end{array} \right\},$$

$$\Delta = \left\{ \begin{array}{l} \text{flies}(X) \leftarrow \text{bird}(X) \\ \neg \text{flies}(X) \leftarrow \text{chicken}(X) \\ \text{flies}(X) \leftarrow \text{chicken}(X), \text{scared}(X) \\ \text{nests\_in\_trees}(X) \leftarrow \text{flies}(X) \end{array} \right\}.$$

In the following examples we abbreviate the above predicates by their first letters, e.g. in the following the predicate  $c/1$  stands for  $\text{chicken}/1$ .

A *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  describes the beliefbase of an agent and therefore contains not all of its beliefs. With the use of strict and defeasible rules it is possible to derive other literals, which may be in the agent's state of belief.

**Definition 3** (Defeasible Derivation). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* and let  $h \in \mathfrak{L}$ . A (*defeasible*) derivation of  $h$  from  $\mathcal{P}$ , denoted  $\mathcal{P} \vdash h$ , consists of a finite sequence  $h_1, \dots, h_n = h$  of literals ( $h_i \in \mathfrak{L}$ ) such that  $h_i$  is a fact ( $h_i \in \Pi$ ) or there exists a strict or defeasible rule in  $\mathcal{P}$  with head  $h_i$  and body  $b_1, \dots, b_k$ , where every  $b_l$  ( $1 \leq l \leq k$ ) is an element  $h_j$  with  $j < i$ . Let  $\mathcal{F}(\mathcal{P})$  denote the set of all literals that have a defeasible derivation from  $\mathcal{P}$ .

If the derivation of a literal  $h$  only uses strict rules, the derivation is called a *strict* derivation.

As facts and strict rules describe strict knowledge, it is reasonable to assume  $\Pi$  to be non-contradictory, i.e. there are no derivations for complementary literals from  $\Pi$  only. But if  $\Pi \cup \Delta$  is contradictory (denoted  $\Pi \cup \Delta \vdash \perp$ ), then there exist defeasible derivations for two complementary literals.

**Definition 4** (Argument, Subargument). Let  $h \in \mathfrak{L}$  be a literal and let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*  $\langle \mathcal{A}, h \rangle$  is an *argument* for  $h$ , iff  $\mathcal{A} \subseteq \Delta$ , there exists a defeasible derivation of  $h$  from  $\mathcal{P}' = (\Pi, \mathcal{A})$ , the set  $\Pi \cup \mathcal{A}$  is non-contradictory and  $\mathcal{A}$  is minimal with respect to set inclusion. The literal  $h$  will be called *conclusion* and the set  $\mathcal{A}$  will be called *support* of the argument  $\langle \mathcal{A}, h \rangle$ . An argument  $\langle \mathcal{B}, q \rangle$  is a *subargument* of an argument  $\langle \mathcal{A}, h \rangle$ , iff  $\mathcal{B} \subseteq \mathcal{A}$ .

**Example 2.** In the *de.l.p.*  $\mathcal{P}$  from Example 1 the literal  $f(tina)$  has the two arguments:  $\langle \{f(tina) \leftarrow b(tina)\}, f(tina) \rangle$  and  $\langle \{f(tina) \leftarrow c(tina), s(tina)\}, f(tina) \rangle$ .

**Definition 5** (Disagreement). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* Two literals  $h$  and  $h_1$  disagree, iff the set  $\Pi \cup \{h, h_1\}$  is contradictory.

Two complementary literals  $p$  und  $\neg p$  disagree trivially, because for every *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  the set  $\Pi \cup \{p, \neg p\}$  is contradictory. But two literals which are not contradictory, can disagree either. For  $\Pi = \{(\neg h \leftarrow b), (h \leftarrow a)\}$  the literals  $a$  and  $b$  disagree, because  $\Pi \cup \{a, b\}$  is contradictory.

**Definition 6** (Counterargument). An argument  $\langle \mathcal{A}_1, h_1 \rangle$  is a *counterargument* to an argument  $\langle \mathcal{A}_2, h_2 \rangle$  at a literal  $h$ , iff there exists a subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$ , such that  $h$  and  $h_1$  disagree.

If  $\langle \mathcal{A}_1, h_1 \rangle$  is a counterargument to  $\langle \mathcal{A}_2, h_2 \rangle$  at a literal  $h$ , then the subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$  is called the *disagreement subargument*. If  $h = h_2$ , then  $\langle \mathcal{A}_1, h_1 \rangle$  is called a *direct attack* on  $\langle \mathcal{A}_2, h_2 \rangle$  and *indirect attack*, otherwise.

**Example 3.** In  $\mathcal{P}$  from Example 1 there is  $\langle \{\neg f(tina) \prec c(tina)\}, \neg f(tina) \rangle$  a direct attack to  $\langle \{f(tina) \prec b(tina)\}, f(tina) \rangle$ . Furthermore  $\langle \{\neg f(tina) \prec c(tina)\}, \neg f(tina) \rangle$  is an indirect attack on  $\langle \{(n(tina) \prec f(tina)), (f(tina) \prec b(tina))\}, n(tina) \rangle$  with the disagreement subargument  $\langle \{(f(tina) \prec b(tina))\}, f(tina) \rangle$ .

A central aspect of defeasible argumentation is a formal comparison criterion among arguments. For some examples of preference criterions see [5]. For the rest of this paper we use an abstract preference criterion  $\succ$  defined as follows.

**Definition 7** (Preference Criterion  $\succ$ ). A preference criterion among arguments is an irreflexive, antisymmetric relation and will be denoted by  $\succ$ . If  $\langle \mathcal{A}_1, h_1 \rangle$  and  $\langle \mathcal{A}_2, h_2 \rangle$  are arguments,  $\langle \mathcal{A}_1, h_1 \rangle$  will be *strictly preferred* over  $\langle \mathcal{A}_2, h_2 \rangle$ , iff  $\langle \mathcal{A}_1, h_1 \rangle \succ \langle \mathcal{A}_2, h_2 \rangle$ .

**Example 4.** A possible preference relation among arguments is *Generalized Specificity* [8]. According to this criterion an argument is preferred to another argument, iff the former one is more *specific* than the latter, i.e. (informally) iff the former one uses more facts or less rules. For example,  $\langle \{c \prec a, b\}, c \rangle$  is more specific than  $\langle \{\neg c \prec a\}, \neg c \rangle$ . For a formal definition see [8,5].

As  $\succ$  is antisymmetric by definition, there cannot be an equipreference among an argument and its counterargument. So we only have to consider the cases, that one argument is better than the other or that two arguments are incomparable with  $\succ$ .

**Definition 8** (Defeater). An argument  $\langle \mathcal{A}_1, h_1 \rangle$  is a *defeater* of an argument  $\langle \mathcal{A}_2, h_2 \rangle$ , iff there is a subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$ , such that  $\langle \mathcal{A}_1, h_1 \rangle$  is a counterargument of  $\langle \mathcal{A}_2, h_2 \rangle$  at literal  $h$  and either  $\langle \mathcal{A}_1, h_1 \rangle \succ \langle \mathcal{A}, h \rangle$  (*proper defeat*) or  $\langle \mathcal{A}_1, h_1 \rangle \not\succ \langle \mathcal{A}, h \rangle$  and  $\langle \mathcal{A}, h \rangle \not\succ \langle \mathcal{A}_1, h_1 \rangle$  (*blocking defeat*).

When considering sequences of arguments, then the definition of defeat is not sufficient to describe a conclusive argumentation line. Defeat only takes an argument and its counterargument into consideration, but disregards preceding arguments. But we expect also properties like *non-circularity* or *concordance* from an argumentation sequence. See [5] for a more detailed description of acceptable argumentation lines.

**Definition 9** (Acceptable Argumentation Line). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* and let  $\Lambda = [\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle]$  be a sequence of arguments.  $\Lambda$  is called *acceptable argumentation line*, iff 1.)  $\Lambda$  is a finite sequence, 2.) every argument  $\langle \mathcal{A}_i, h_i \rangle$  with  $i > 1$  is a defeater of his predecessor  $\langle \mathcal{A}_{i-1}, h_{i-1} \rangle$  and if  $\langle \mathcal{A}_i, h_i \rangle$  is a blocking defeater of  $\langle \mathcal{A}_{i-1}, h_{i-1} \rangle$  and  $\langle \mathcal{A}_{i+1}, h_{i+1} \rangle$  exists, then  $\langle \mathcal{A}_{i+1}, h_{i+1} \rangle$  is a proper

defeater of  $\langle \mathcal{A}_i, h_i \rangle$ , 3.)  $\Pi \cup \mathcal{A}_1 \cup \mathcal{A}_3 \cup \dots$  is non-contradictory (*concordance of supporting arguments*), 4.)  $\Pi \cup \mathcal{A}_2 \cup \mathcal{A}_4 \cup \dots$  is non-contradictory (*concordance of interfering arguments*), and 5.) no argument  $\langle \mathcal{A}_k, h_k \rangle$  is a subargument of an argument  $\langle \mathcal{A}_i, h_i \rangle$  with  $i < k$ .

Let  $+$  denote the concatenation of argumentation lines and arguments, e.g.  $[\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle] + \langle \mathcal{B}, h \rangle$  stands for  $[\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle, \langle \mathcal{B}, h \rangle]$ .

In DeLP a literal  $h$  is *warranted*, if there exists an argument  $\langle \mathcal{A}, h \rangle$  which is non-defeated in the end. To decide whether  $\langle \mathcal{A}, h \rangle$  is defeated or not, every acceptable argumentation line starting with  $\langle \mathcal{A}, h \rangle$  has to be considered.

**Definition 10** (Dialectical Tree). Let  $\langle \mathcal{A}_0, h_0 \rangle$  be an argument of a *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$ . A *dialectical tree* for  $\langle \mathcal{A}_0, h_0 \rangle$ , denoted  $\mathcal{T}_{\langle \mathcal{A}_0, h_0 \rangle}$ , is defined by

1. The root of  $\mathcal{T}$  is  $\langle \mathcal{A}_0, h_0 \rangle$ .
2. Let  $\langle \mathcal{A}_n, h_n \rangle$  be a node in  $\mathcal{T}$  and let  $\Lambda = [\langle \mathcal{A}_0, h_0 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle]$  be the sequence of nodes from the root to  $\langle \mathcal{A}_n, h_n \rangle$ . Let  $\langle \mathcal{B}_1, q_1 \rangle, \dots, \langle \mathcal{B}_k, q_k \rangle$  be the defeaters of  $\langle \mathcal{A}_n, h_n \rangle$ . For every defeater  $\langle \mathcal{B}_i, q_i \rangle$  with  $1 \leq i \leq k$ , such that the argumentation line  $\Lambda' = [\langle \mathcal{A}_0, h_0 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle, \langle \mathcal{B}_i, q_i \rangle]$  is acceptable, the node  $\langle \mathcal{A}_n, h_n \rangle$  has a child  $\langle \mathcal{B}_i, q_i \rangle$ . If there is no such  $\langle \mathcal{B}_i, q_i \rangle$ , the node  $\langle \mathcal{A}_n, h_n \rangle$  is a leaf.

In order to decide whether the argument at the root of a given dialectical tree is defeated or not, it is necessary to perform a *bottom-up*-analysis of the tree. There every leaf of the tree is marked “*undefeated*” and every inner node is marked “*defeated*”, if it has at least one child node marked “*undefeated*”. Otherwise it is marked “*undefeated*”. Let  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}^*$  denote the marked dialectical tree of  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}$ .

**Definition 11** (Warrant). A literal  $h \in \mathfrak{L}$  is *warranted*, iff there exists an argument  $\langle \mathcal{A}, h \rangle$  for  $h$ , such that the root of the marked dialectical tree  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}^*$  is marked “*undefeated*”. Then  $\langle \mathcal{A}, h \rangle$  is a *warrant* for  $h$ .

If a literal  $h$  is a fact or has a strict derivation from a *de.l.p.*, then  $h$  is also warranted as there are no counterarguments for  $\langle \emptyset, h \rangle$ . Based on the notion of warrant, the answer behaviour of a DeLP-interpreter can be defined as follows.

**Definition 12** (Answers to queries). The answer of a DeLP-interpreter to a query  $h$  is defined as 1.) YES, iff  $h$  is warranted, 2.) NO, iff  $\bar{h}$  is warranted, 3.) UNDECIDED, iff neither  $h$  nor  $\bar{h}$  are warranted and 4.) UNKNOWN, iff  $h \notin \mathfrak{L}$ .

### 3. Using Defeasible Logic Programming For a Distributed Environment

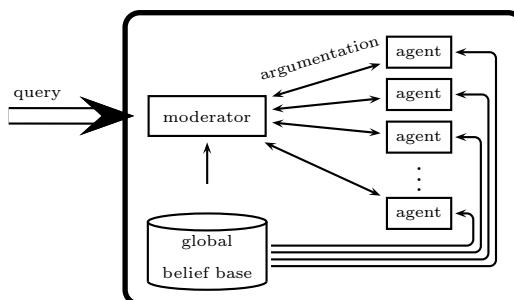
In an argumentation-based multi agent system (**ArgMAS**) several agents argue with each other about the truth value of a given logical sentence. In contrast to general DeLP the agents do not have knowledge about the beliefs of other agents and only react on their arguments. So as an **ArgMAS** consists of several components, the belief of an **ArgMAS** is divided among the components. Some belief can be seen as strict knowledge, that should be shared among all agents

of the system, e.g. knowledge about the current law or general facts as “every penguin is a bird”. But every agent possesses also some individual beliefs like preferences or personal opinions, e.g. “if  $X$  is a gardener, then  $X$  is rather the murderer than anyone else”. Thus the belief in an ArgMAS is divided into a *global belief base* and several *local belief bases*. An important constraint on belief bases is consistency. As the global belief base represents strict knowledge, it should be consistent in itself. Furthermore every local belief base should be consistent with the global belief base, as every agent’s belief should not contradict with common knowledge, e.g. an agent should not argue that *john* is the murderer, if “*john* has an alibi” is strict knowledge. But as the opinions of different agents can differ, the union of all local belief bases and the global belief base can be inconsistent. The formal definition of belief bases will be given below.

### 3.1. Overview

An ArgMAS takes a literal as a query, generates arguments with an internal deliberation mechanism and then returns a statement about acceptance or disapproval.

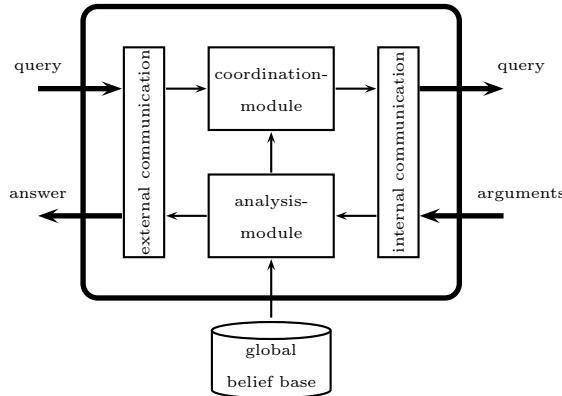
In order to make a set of agents interact in a cooperative manner a coordination mechanism is needed. We use the centralized approach of *organisational structuring* [6] to get the external communications of the systems separated from the internal deliberation and coordination. A special agent, called *moderator*, will be used as an interface of the system to the outside world and as contact for queries. Furthermore the moderator coordinates the argumentation process between the other agents and finally analyzes the resulting argumentation structures to come up with an answer to the given query. Figure 1 shows a pattern of the message transfer in an ArgMAS and the special role of the moderator. The global belief base consists of the common knowledge of the whole system.



**Figure 1.** An argumentation-based multi agent system (ArgMAS)

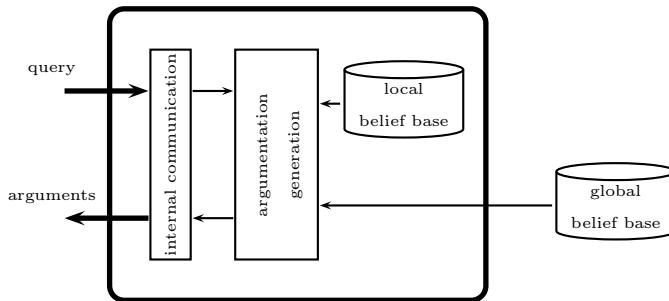
The moderator of an ArgMAS must accept literals as queries from the outside world and be able to return answers. Furthermore he must send and receive internal queries to and from other agents and analyze the resulting dialectical trees of the actual argumentation process. Figure 2 shows the components of a moderator.

The actual argumentation process is done by the agents. They generate arguments on the basis of the global and their particular local belief bases and react



**Figure 2.** The internal components of a moderator

on arguments of other agents with counterarguments. An agent must be capable of inferring new beliefs by using the defeasible rules of his local belief base and the strict rules of the global belief base. Based on these inferences he generates suitable counterarguments to arguments from other agents. As the proposed system is centralized an agent only has to communicate with the moderator and the latter arbitrates between the individual agents. Figure 3 shows the components of an agent capable of argumentation. Notice that every agent has access to the global belief base and also has a local belief base.



**Figure 3.** An argumentation-capable agent

In the next subsection we will formalize this argumentation process, the moderator and the agents.

### 3.2. Formalization

As stated above the belief of an ArgMAS is divided into global and local belief bases. While the global belief base contains strict knowledge and therefore is built of facts and strict rules, the local belief bases are assumed to comprise only defeasible rules.

**Definition 13** (Belief bases). A *global belief base*  $\Pi$  is a non-contradictory set of strict rules and facts. A set of defeasible rules  $\Delta$  is called a *local belief base relative to*  $\Pi$ , if  $\Pi$  is a global belief base and  $\Pi \cup \Delta$  is non-contradictory.

Besides the data structures for knowledge representation the components of an ArgMAS consist of functions to realize argumentation generation and analysis. The moderator must be capable of checking argumentation sequences for acceptance and evaluation of dialectical trees. Therefore in this section the necessary functions will be formally specified.

As in Section 2 the symbol  $\mathfrak{L}$  denotes the set of all literals that may appear in a belief base. Furthermore let  $\mathcal{R}$  be the set of all defeasible rules, that can be constructed with literals from  $\mathfrak{L}$ ,  $\Omega$  the set of all possible arguments that can be built using rules from  $\mathcal{R}$  and conclusions from  $\mathfrak{L}$ ,  $\Sigma$  the set of all sequences of arguments from  $\Omega$  and  $\Upsilon$  the set of all dialectical trees of arguments from  $\Omega$ .

We start with formal description of the functional components of a moderator.

**Definition 14** (Analysis function). An *analysis function*  $\chi$  is a function  $\chi : \Upsilon \rightarrow \{0, 1\}$ , such that for every dialectical tree  $v \in \Upsilon$  it holds  $\chi(v) = 1$  iff the root argument of  $v$  is undefeated.

The definition of an analysis function is independent of the definition of dialectical trees.

**Example 5.** Let  $v \in \Upsilon$  be a dialectical tree according to Definition 10 and let  $v^*$  be the corresponding marked dialectical tree. Then the analysis function  $\chi_D$  is defined as  $\chi_D(v) = 1$  iff the root of  $v^*$  “undefeated” and  $\chi_D(v) = 0$  otherwise.

An acceptance function tests a given argument sequence for acceptance.

**Definition 15** (Acceptance function). An *acceptance function*  $\eta$  is a function  $\eta : \Sigma \rightarrow \{0, 1\}$ , such that for every argument sequence  $\Lambda \in \Sigma$  it holds  $\eta(\Lambda) = 1$  iff  $\Lambda$  is accepted.

**Example 6.** Let  $\succ$  be a preference relation among arguments. Then the acceptance function  $\eta_{D,\succ}$  is defined as  $\eta_{D,\succ}(\Lambda) = 1$  iff  $\Lambda$  is acceptable with respect to  $\succ$  and  $\eta_{D,\succ}(\Lambda) = 0$  otherwise.

Let  $\mathfrak{P}(S)$  denote the power set of  $S$ .

**Definition 16** (Decision function). A *decision function*  $\mu$  is a function  $\mu : \mathfrak{P}(\Upsilon) \rightarrow \{\text{YES}, \text{NO}, \text{UNDECIDED}, \text{UNKNOWN}\}$ .

A decision function  $\mu$  maps a set of dialectical trees to the suitable answer of the system. It is sufficient to define  $\mu$  only on sets of dialectical trees with root arguments for or against the same atom as no other dialectical tree can be generated in an argumentation process for this particular atom.

**Example 7.** Let the moderator decide the answer to a query  $p$  on the basis of Definition 12. Let  $Q \subseteq \Upsilon$  such that all root arguments of dialectical trees in  $Q$  are arguments for  $p$  or for  $\bar{p}$ , then the decision function  $\mu_D$  is defined as

1.  $\mu_D(Q) = \text{YES}$ , if there exists a dialectical tree  $v \in Q$  s.t. the root of  $v$  is an argument for  $p$  and  $\chi_D(v) = 1$ .
2.  $\mu_D(Q) = \text{NO}$ , if there exists a dialectical tree  $v \in Q$  s.t. the root of  $v$  is an argument for  $\bar{p}$  and  $\chi_D(v) = 1$ .
3.  $\mu_D(Q) = \text{UNDECIDED}$ , if  $\chi_D(v) = 0$  for all  $v \in Q$ .
4.  $\mu_D(Q) = \text{UNKNOWN}$ , if  $p$  is not in the language ( $p \notin \mathcal{L}$ ).

**Definition 17** (Moderator). A *moderator* is a tuple  $(\mu, \chi, \eta)$  with a decision function  $\mu$ , an analysis function  $\chi$  and an acceptance function  $\eta$ . A moderator  $(\mu, \chi, \eta)$  is called a *DeLP-moderator*, if  $\mu = \mu_D$ ,  $\chi = \chi_D$  and  $\eta = \eta_{D,\succ}$  for a given preference relation  $\succ$  among arguments.

An agent of an ArgMAS has to provide two functions: On the one hand he must be capable of generating initial arguments based on given literals and on the other hand he must be capable of generating counterarguments to arguments given by other agents.

**Definition 18** (Root argument function). Let  $\Pi$  be a global belief base and let  $\Delta$  be a local belief base. A *root argument function*  $\varphi_{\Pi,\Delta}$  relative to  $\Pi$  and  $\Delta$  is a function  $\varphi_{\Pi,\Delta} : \mathcal{L} \rightarrow \mathfrak{P}(\Omega)$ , such that for every literal  $h \in \mathcal{L}$  the set  $\varphi_{\Pi,\Delta}(h)$  is a set of arguments for  $h$  or for  $\bar{h}$  from  $\Pi$  and  $\Delta$ .

To guarantee autonomy in argument generation of an agent, every agent has an own acceptance function  $\eta$  to test his generated arguments on acceptance in the current argument sequence. On the basis of  $\eta$  the counterargument function is defined as follows

**Definition 19** (Counterargument function). Let  $\Pi$  be a global belief base, let  $\Delta$  be a local belief base, and let  $\eta$  be an acceptance function. A *counterargument function*  $\psi_{\Pi,\Delta}$  relative to  $\Pi$  and  $\Delta$  is a function  $\psi_{\Pi,\Delta} : \Sigma \rightarrow \mathfrak{P}(\Omega)$ , such that for every argumentation sequence  $\Lambda \in \Sigma$  the set  $\psi_{\Pi,\Delta}(\Lambda)$  is a set of attacks from  $\Pi$  and  $\Delta$  on the last argument of  $\Lambda$  from  $\Omega$  and for every  $\langle \mathcal{B}, h \rangle \in \psi_{\Pi,\Delta}(\Lambda)$  it holds that  $\eta(\Lambda + \langle \mathcal{B}, h \rangle) = 1$ .

An agent of an ArgMAS is then defined as:

**Definition 20** (Agent). Let  $\Pi$  be a global belief base. An *agent* relative to  $\Pi$  is a tuple  $(\Delta, \varphi_{\Pi,\Delta}, \psi_{\Pi,\Delta}, \eta)$  with a local belief base  $\Delta$  relative to  $\Pi$ , a root argument function  $\varphi_{\Pi,\Delta}$ , a counterargument function  $\psi_{\Pi,\Delta}$  and an acceptance function  $\eta$ .

In the following we omit the subscripts  $\Pi$  and  $\Delta$  for root argument and counterargument functions when they are clear from context.

Putting things together, we define an argumentation-based multi agent system to consist of one moderator, a global belief base and a set of agents:

**Definition 21** (Argumentation-based multi agent system). An *argumentation-based multi agent system* (ArgMAS) is a tuple  $(M, \Pi, \{A_1, \dots, A_n\})$  with a moderator  $M$ , a global belief base  $\Pi$  and agents  $A_1, \dots, A_n$  relative to  $\Pi$ .

We now develop a functional description of the actual argumentation process to determine the answer to a query  $h \in \mathcal{L}$ .

**Definition 22** (Argumentation product). Let  $h \in \mathcal{L}$  be a query and  $T = (M, \Pi, \{A_1, \dots, A_n\})$  an ArgMAS with  $M = (\mu, \chi, \eta)$  and  $A_i = (\Delta_i, \varphi_i, \psi_i, \eta_i)$  for  $1 \leq i \leq n$ . A dialectical tree  $v$  is called *argumentation product* of  $T$  and  $h$ , iff the following conditions hold: 1.) there exists a  $j$  with  $1 \leq j \leq n$ , such that the root of  $v$  is an element of  $\varphi_j(h)$ , and 2.) for every path  $\Lambda = [\langle A_1, h_1 \rangle, \dots, \langle A_n, h_n \rangle]$  in  $v$  and the set  $K$  of child nodes of  $\langle A_n, h_n \rangle$  it holds  $K = \{\langle B, h' \rangle | \langle B, h' \rangle \in \psi_1(\Lambda) \cup \dots \cup \psi_n(\Lambda) \text{ and } \eta(\Lambda + \langle B, h' \rangle) = 1\}$  ( $K$  is the set of all acceptable attacks on  $\Lambda$ ).

The answer behaviour of an ArgMAS is based on the argumentation products and the decision function of the moderator:

**Definition 23** (Answer behaviour). Let  $h \in \mathcal{L}$  be a query,  $T = (M, \Delta, \{A_1, \dots, A_n\})$  an ArgMAS with  $M = (\mu, \chi, \eta)$  and let  $\{v_1, \dots, v_n\}$  be the set of all argumentation products of  $T$  and  $h$ . The answer  $A(T, h)$  of  $T$  on the query  $h$  is  $A(T, h) = \mu(\{v_1, \dots, v_n\})$ .

The possible answers of an ArgMAS to a query are therefore YES, NO, UNDECIDED and UNKNOWN as it is for general DeLP.

#### 4. Comparison with general Defeasible Logic Programming

In this section we compare the distributed framework for defeasible argumentation with general DeLP. As the definition of the components of an ArgMAS are based upon DeLP the comparison is realized via translation of an DeLP-program into an ArgMAS. We will show, that the answer behaviour of the original system and its translated counterpart will remain the same in most cases. All proofs of the following theorems can be found in an extended version of this paper [10].

For this section, the acceptance functions of all agents are identical with the acceptance function of the moderator, as this is also the situation in ordinary DeLP. Furthermore the root argument and counterargument functions of all agents are *maximal*, i. e. these functions always return the maximal set of possible arguments with respect to their beliefs.

To translate a DeLP-program  $\mathcal{P}$  into an ArgMAS and maintain consistent belief bases, the set of defeasible rules of  $\mathcal{P}$  have to be appropriately divided among several agents. To sustain identical answer behaviours a very naive translation will suffice.

**Definition 24** (Argument-based rule-division). Let  $\mathcal{P}$  be a *de.l.p.* and  $Q$  the set of all arguments of  $\mathcal{P}$ . Then the *argument-based rule-division*  $AD(\mathcal{P})$  of  $\mathcal{P}$  is defined by  $AD(\mathcal{P}) = \{\mathcal{A} | \langle \mathcal{A}, h \rangle \in Q\}$ .

For every possible argument  $\langle \mathcal{A}, h \rangle$  there will be one agent with a local belief base  $\mathcal{A}$ . As arguments are consistent by definition, every agent has a consistent belief base. Furthermore no argument gets lost by translation of a DeLP into an ArgMAS.

**Definition 25** ( $\mathcal{P}$ -Induced ArgMAS). Let  $AD(\mathcal{P}) = \{\Delta_1, \dots, \Delta_n\}$  the argument-based rule-devision of a *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$ . Let  $A_1, \dots, A_n$  be agents with local belief bases  $\Delta_1, \dots, \Delta_n$  respectively and let  $M$  be a DeLP-moderator. Then  $T = (M, \Pi, \{A_1, \dots, A_n\})$  is the  $\mathcal{P}$ -induced ArgMAS.

As the local belief bases of the agents  $A_1, \dots, A_n$  are consistent by construction, every  $\mathcal{P}$ -induced ArgMAS is indeed an ArgMAS according to Definition 21. Furthermore the answer behaviour of the  $\mathcal{P}$ -induced ArgMAS is identical with the answer behaviour of  $\mathcal{P}$  given identical preference relations for arguments.

**Theorem 1.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* und let  $T = (M, \Pi, \{A_1, \dots, A_n\})$  be the  $\mathcal{P}$ -induced ArgMAS. Let  $h$  be a query to  $\mathcal{P}$  and  $A$  the answer of  $\mathcal{P}$ , e.g.  $A \in \{\text{YES}, \text{NO}, \text{UNDECIDED}, \text{UNKNOWN}\}$ . Then  $A$  is also the answer of  $T$  to  $h$ .

So every *de.l.p.* can be transformed into an ArgMAS while preserving its answer behaviour. The converse naive translation of an ArgMAS into a *de.l.p.* is not so easily possible without some restrictions.

**Definition 26** ( $T$ -induced *de.l.p.*). Let  $T = (M, \Pi, \{A_1, \dots, A_n\})$  be an ArgMAS with a DeLP-moderator  $M$  and let  $\Delta_1, \dots, \Delta_n$  be the local belief base of agents  $A_1, \dots, A_n$ . Then  $\mathcal{P} = (\Pi, \Delta_1 \cup \dots \cup \Delta_n)$  is called the  $T$ -induced *de.l.p.*.

For arbitrary local belief bases  $\Delta_1, \dots, \Delta_n$  the answer behaviour of  $T$  and the  $T$ -induced *de.l.p.* is not necessarily the same as the following example shows.

**Example 8.** Let  $T = (M, \Pi, \{A_1, A_2\})$  be an ArgMAS and let  $\Delta_1$  and  $\Delta_2$  be the local belief bases of  $A_1$  and  $A_2$  with  $\Delta_1 = \{(b \prec a), (b \prec a, c)\}$  and  $\Delta_2 = \{(\neg b \prec a), (c \prec d)\}$  and let furthermore  $\Pi = \{a, d\}$ . Given the query  $b$ ,  $T$  yields two argumentation products  $[\langle\{(b \prec a)\}, b\rangle, \langle\{(\neg b \prec a)\}, \neg b\rangle]$  and  $[\langle\{(\neg b \prec a)\}, \neg b\rangle, \langle\{(b \prec a)\}, b\rangle]$ . As the roots of both argumentation products will be marked “defeated”, the answer of  $T$  on  $b$  is UNDECIDED.

The  $T$ -induced *de.l.p.*  $\mathcal{P} = (\Pi', \Delta')$  is given by  $\Pi' = \{a, d\}$  and  $\Delta = \{(b \prec a), (b \prec a, c), (\neg b \prec a), (c \prec d)\}$  and yields on the query  $b$  among others the argumentation product  $[\langle\{(b \prec a)\}, b\rangle, \langle\{(\neg b \prec a)\}, \neg b\rangle, \langle\{(b \prec a, c), (c \prec d)\}, b\rangle]$ . As there the root will be marked with “undefeated”, the answer of  $\mathcal{P}$  on  $b$  is YES.

The reason for the different answer behaviour of  $T$  and  $\mathcal{P}$  in Example 8 is the “misplaced” rule  $c \prec d$ , which can not be used for any argument on the query  $b$  by  $A_2$ . By forbidding “misplaced” rules in an ArgMAS, a translation into a *de.l.p.* with protection of answer behaviour is possible. This yields the notion of a well-formed ArgMAS.

**Definition 27** (Well-formed ArgMAS). Let  $T = (M, \Pi, \{A_1, \dots, A_n\})$  be an ArgMAS and let  $\Delta_1, \dots, \Delta_n$  the local belief bases of agents  $A_1, \dots, A_n$ . Let furthermore  $\mathcal{P}$  be the  $T$ -induced *de.l.p.*.  $T$  is called a *well-formed* ArgMAS iff for every argument  $\langle \mathcal{A}, h \rangle$  in  $\mathcal{P}$  there exist an  $i$  ( $1 \leq i \leq n$ ) with  $\mathcal{A} \subseteq \Delta_i$ .

**Theorem 2.** Let  $T = (M, \Pi, \{A_1, \dots, A_n\})$  be a well-formed ArgMAS and let  $\Delta_1, \dots, \Delta_n$  be the local belief bases of agents  $A_1, \dots, A_n$ . Let furthermore  $\mathcal{P}$  be

the  $T$ -induced de.l.p.. If  $h$  is a query and  $A$  the answer of  $T$  to  $h$ , then  $A$  is also the answer of  $\mathcal{P}$  to  $h$ .

As every *de.l.p.* can be translated in an ArgMAS without losing information, distributed defeasible argumentation as proposed in this paper can be seen as a generalization of ordinary defeasible argumentation.

## 5. Remarks and Conclusion

The examination of distributed argumentation leads to new insights into logic-based argumentation in AI and discloses new application areas. The approach of distributed argumentation proposed in this paper distinguishes explicitly between several entities with different opinions on a structural basis. The proposed framework was implemented for a diploma thesis and a complex legal dispute was realised to illustrate the concept [9]. In that example two agents took the roles of accuser and defender, respectively, and argue about a legal claim. Parts of german law were translated to support the arguments and counterarguments of the two agents.

As the results in Section 4 show the proposed framework can subsume general defeasible logic programming and therefore is compatible with the existing theory on DeLP. As part of our ongoing work we plan to generalize the proposed system with the use of abstract argumentation frameworks [4] and investigate relationships of distributed argumentation using DeLP with game theory [3].

**Acknowledgments** The authors thank the reviewers for their helpful comments to improve the original version of this paper.

## References

- [1] Leila Amgoud, Yannis Dimopoulos, and Pavlos Moraitis. A unified and general framework for argumentation-based negotiation. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multi-Agents Systems, AAMAS'2007*, May 2007.
- [2] T.J.M. Bench-Capon. Persuasion in practical argument using value based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
- [3] Laura A. Cecchi, Pablo R. Fillottrani, and Guillermo R. Simari. On the complexity of DeLP through game semantics. In J. Dix and A. Hunter, editors, *Proc. 11th Intl. Workshop on Nonmonotonic Reasoning (NMR 2006)*, pages 386–394, Windermere, UK, 2006.
- [4] Phan M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *AI Journal*, 77(2):321–358, 1995.
- [5] A. García and G. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1-2):95–138, 2002.
- [6] H. S. Nwana, L. C. Lee, and N. R. Jennings. Coordination in software agent systems. *The British Telecom Technical Journal*, 14(4):79–88, 1996.
- [7] Simon Parsons, Carles Sierra, and Nick Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.
- [8] F. Stolzenburg, A. García, C. Chesñevar, and G. Simari. Computing generalized specificity. *Journal of Non-Classical Logics*, 13(1):87–113, 2003.
- [9] M. Thimm. *Verteilte logikbasierte Argumentation: Konzeption, Implementierung und Anwendung im Rechtswesen*. VDM Verlag Dr. Müller, 2008.
- [10] M. Thimm and G. Kern-Isberner. A distributed argumentation framework using defeasible logic programming (extended version). Technical report, TU Dortmund, 2008.

# On the Relationship of Defeasible Argumentation and Answer Set Programming

Matthias Thimm<sup>a</sup> Gabriele Kern-Isberner<sup>a</sup>

<sup>a</sup> *Information Engineering Group, Faculty of Computer Science  
Technische Universität Dortmund, Germany*

**Abstract.** This paper investigates the relationship between defeasible argumentation (DeLP) and answer set programming by transforming a defeasible logic program into an answer set program. We propose two types of conversions that differ with respect to the handling of strict rules. Inference via a dialectical warrant procedure in DeLP turns out to be stronger than credulous answer set inference in both cases, while conversions of the second type bring DeLP inference closer to skeptical answer set inference. Moreover, we investigate some characteristics of the warrant procedure of DeLP which lead to a better understanding of the notion of warrant.

**Keywords.** Argumentation, defeasible logic programming, answer set programming

## 1. Introduction

*Defeasible Argumentation* [8], as proposed with the language DeLP (*Defeasible Logic Programming*) by García and Simari in [6] is an approach for logical argumentative reasoning [1,9] based on defeasible logic. In DeLP the belief in literals is supported by arguments and in order to handle conflicting information a warrant procedure decides which information has the strongest grounds to believe in. In this way, the notion of warrant induces a nonmonotonic inference relation between a defeasible logic program (consisting of facts as well as strict and defeasible rules) and literals. The exploration of this inference relation in terms of answer set semantics is the topic of this paper.

Indeed, the relationships between defeasible argumentation and other default reasoning systems, especially the relationship of their particular inference mechanisms, have been investigated only little so far. While in [5] default logic and logic programming are characterized as instantiations of Dung's abstract argumentation framework we are interested in a direct relation between default logic and DeLP which can also be characterized as an instantiation of an abstract argumentation framework. In [4] the relationship of DeLP with Reiters default logic [10] is investigated by converting a default logic program into a defeasible logic program and applying the warrant procedure to determine the extensions of the original

default logic program. In that paper, a special case of DeLP programs is used, so that the warrant of a literal is equivalent to the sceptical inference of that literal.

In this paper we take the converse point of view by translating a defeasible logic program into an answer set program (ASP) [7] and applying answer set techniques to determine the warranted literals of the original defeasible logic program. First, we investigate some characteristics of the warrant procedure of DeLP which leads to a better understanding of the notion of warrant. As DeLP reasoning is paraconsistent, the handling of inconsistencies under the translation is of major importance. We will propose two approaches to converting a defeasible logic program into an answer set program, dealing with inconsistencies in different ways. The first conversion method respects the substantial difference between strict and defeasible rules but has to take inconsistencies brought about by strict rules into account; the resulting warrant semantics is shown to be weaker than skeptical ASP semantics, but stronger than credulous ASP semantics. The other type of conversion blurs the distinction between strict and default rules and yields better results in computing warrant through answer set techniques. More precisely, we show that all warranted literals are contained in one answer set of the corresponding logic program. In particular, if the preference relation between arguments is empty (so that defeating is reduced to attacking), then inference by a warrant procedure turns out to be even stronger than skeptical inference.

In contrast to [2] this paper does not aim at fixing DeLP regarding some observed flaws in its inference mechanism; instead, we will interpret the original DeLP inference mechanism via answer set semantics.

This paper is structured as follows: in Section 2 and 3 brief overviews over ASP and defeasible logic programming are given. Section 4 investigates the notion of warrant in detail. Section 5 and 6 propose two alternatives of converting a DeLP-program into an answer set program and discuss the results. In Section 7 we conclude. All proofs can be found in an extended version of this paper [12].

## 2. Answer set programming

In this section we give a brief overview over answer set programming and answer sets as proposed by Gelfond and Lifschitz in [7]. We consider extended logic programs, which distinguish between classical and default negation.

We use a first-order language without function symbols except constants, so let  $\mathcal{L}$  be a set of literals, where a literal  $h$  is an atom  $A$  or a (classical) negated atom  $\neg A$ . The symbol  $\bar{\cdot}$  will be used to denote the complement of a literal with respect to classical negation, i.e. it is  $\bar{p} = \neg p$  and  $\bar{\neg p} = p$  for a ground atom  $p$ .

**Definition 1** (Extended logic program). An *extended logic program*  $P$  is a finite set of rules of the form  $r : h \leftarrow a_1, \dots, a_n, \text{not } b_1, \dots, \text{not } b_m$  where  $h, a_1, \dots, a_n, b_1, \dots, b_m \in \mathcal{L}$ . We denote by  $\text{head}(r)$  the head  $h$  of the rule  $r$  and by  $\text{body}(r)$  the body  $\{a_1, \dots, a_n, \text{not } b_1, \dots, \text{not } b_m\}$  of the rule  $r$ .

If the body of a rule  $r$  is empty ( $\text{body}(r) = \emptyset$ ), then  $r$  is called a *fact*, abbreviated  $h$  instead of  $h \leftarrow$ .

Given a set  $X \subseteq \mathfrak{L}$  of literals, then  $r$  is *applicable* in  $X$ , iff  $a_1, \dots, a_n \in X$  and  $b_1, \dots, b_m \notin X$ . The rule  $r$  is *satisfied* by  $X$ , if  $h \in X$  or if  $r$  is not applicable in  $X$ .  $X$  is a model of an extended logic program  $P$  iff all rules of  $P$  are satisfied by  $X$ . The set  $X \subseteq \mathfrak{L}$  is *consistent*, iff for every  $h \in X$  it is not the case that  $\bar{h} \in X$ . An answer set is a minimal consistent set of literals that satisfies all rules. This can be characterized as follows.

**Definition 2** (Reduct). Let  $P$  be an extended logic program and  $X \subseteq \mathfrak{L}$  a set of literals. The  $X$ -reduct of  $P$ , denoted  $P^X$ , is the union of all rules  $h \leftarrow a_1, \dots, a_n$  such that  $h \leftarrow a_1, \dots, a_n, \text{not } b_1, \dots, \text{not } b_m \in P$  and  $X \cap \{b_1, \dots, b_m\} = \emptyset$ .

For any extended logic program  $P$  and a set  $X$  of literals, the  $X$ -reduct of  $P$  is a logic program  $P'$  without default-negation and therefore has a minimal model. If  $P'$  is inconsistent, then its unique model is defined to be  $\mathfrak{L}$ .

**Definition 3** (Answer set). Let  $P$  be an extended logic program. A consistent set of literals  $S \subseteq \mathfrak{L}$  is an *answer set* of  $P$ , iff  $S$  is a minimal model of  $P^S$ .

### 3. Defeasible Logic Programming

Defeasible Logic Programming (DeLP) [6] is a logic programming language which is capable of modelling defeasible knowledge. With the use of a defeasible argumentation process it is possible to derive conclusive knowledge.

The basic elements of DeLP are facts and rules. The set of rules is divided into strict rules, i. e. rules which derive certain knowledge, and defeasible rules, i. e. rules which derive uncertain or defeasible knowledge. We use the same set  $\mathfrak{L}$  of literals as in Section 2 to define the elements of a DeLP-program.

**Definition 4** (Fact, strict rule, defeasible rule). A *fact* is a literal  $h \in \mathfrak{L}$ . A *strict rule* is an ordered pair  $h \leftarrow B$ , where  $h \in \mathfrak{L}$  and  $B \subseteq \mathfrak{L}$ . A *defeasible rule* is an ordered pair  $h \prec B$ , where  $h \in \mathfrak{L}$  and  $B \subseteq \mathfrak{L}$ .

As in ASP we use the functions *body/1* and *head/1* to refer to the head resp. body of a defeasible or strict rule.

**Definition 5** (Defeasible Logic Program). A *Defeasible Logic Program*  $\mathcal{P} = (\Pi, \Delta)$ , abbreviated *de.l.p.*, consists of a (possibly infinite) set  $\Pi$  of facts and strict rules and of a (possibly infinite) set  $\Delta$  of defeasible rules.

**Example 1** ([6], example 2.1). Let  $\mathcal{P} = (\Pi, \Delta)$  be given by

$$\begin{aligned} \Pi &= \left\{ \begin{array}{ll} \text{chicken}(tina) & \text{scared}(tina) \\ \text{penguin}(tweety) & (\text{bird}(X) \leftarrow \text{chicken}(X)) \\ \text{bird}(X) \leftarrow \text{penguin}(X) & (\neg \text{flies}(X) \leftarrow \text{penguin}(X)) \end{array} \right\}, \\ \Delta &= \left\{ \begin{array}{l} \text{flies}(X) \prec \text{bird}(X) \\ \neg \text{flies}(X) \prec \text{chicken}(X) \\ \text{flies}(X) \prec \text{chicken}(X), \text{scared}(X) \\ \text{nests\_in\_trees}(X) \prec \text{flies}(X) \end{array} \right\}. \end{aligned}$$

In the following examples we abbreviate the above predicates by their first letters, e.g. in the following the predicate  $c/1$  stands for *chicken/1*.

A *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  describes the belief base of an agent and therefore contains not all of its beliefs. With the use of strict and defeasible rules it is possible to derive other literals, which may be in the agent's state of belief.

**Definition 6** (Defeasible Derivation). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* and let  $h \in \mathfrak{L}$ . A (*defeasible*) derivation of  $h$  from  $\mathcal{P}$ , denoted  $\mathcal{P} \succsim h$ , consists of a finite sequence  $h_1, \dots, h_n = h$  of literals ( $h_i \in \mathfrak{L}$ ) such that  $h_i$  is a fact ( $h_i \in \Pi$ ) or there exists a strict or defeasible rule in  $\mathcal{P}$  with head  $h_i$  and body  $b_1, \dots, b_k$ , where every  $b_l$  ( $1 \leq l \leq k$ ) is an element  $h_j$  with  $j < i$ . Let  $\mathcal{F}(\mathcal{P})$  denote the set of all literals that have a defeasible derivation from  $\mathcal{P}$ .

If the derivation of a literal  $h$  only uses strict rules, the derivation is called a *strict* derivation.

As facts and strict rules describe strict knowledge, it is reasonable to assume  $\Pi$  to be non-contradictory, i.e. there are no derivations for complementary literals from  $\Pi$  only. But if  $\Pi \cup \Delta$  is contradictory (denoted  $\Pi \cup \Delta \succsim \perp$ ), then there exist defeasible derivations for complementary literals.

**Definition 7** (Argument, Subargument). Let  $h \in \mathfrak{L}$  be a literal and let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*  $\langle \mathcal{A}, h \rangle$  is an *argument* for  $h$ , iff  $\mathcal{A} \subseteq \Delta$ , there exists a defeasible derivation of  $h$  from  $\mathcal{P}' = (\Pi, \mathcal{A})$ , the set  $\Pi \cup \mathcal{A}$  is non-contradictory and  $\mathcal{A}$  is minimal with respect to set inclusion. The literal  $h$  will be called *conclusion* and the set  $\mathcal{A}$  will be called *support* of the argument  $\langle \mathcal{A}, h \rangle$ . An argument  $\langle \mathcal{B}, q \rangle$  is a *subargument* of an argument  $\langle \mathcal{A}, h \rangle$ , iff  $\mathcal{B} \subseteq \mathcal{A}$ .

**Example 2.** In the *de.l.p.*  $\mathcal{P}$  from Example 1  $f(tina)$  has the two arguments  $\langle \{f(tina) \negleftarrow b(tina)\}, f(tina) \rangle$  and  $\langle \{f(tina) \negleftarrow c(tina), s(tina)\}, f(tina) \rangle$ .

**Definition 8** (Disagreement). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*. Two literals  $h$  and  $h_1$  *disagree*, iff the set  $\Pi \cup \{h, h_1\}$  is contradictory.

Two complementary literal  $p$  und  $\neg p$  disagree trivially, but two literals which are not contradictory, can disagree either. For  $\Pi = \{(\neg h \leftarrow b), (h \leftarrow a)\}$  the literals  $a$  and  $b$  disagree, because  $\Pi \cup \{a, b\}$  is contradictory.

**Definition 9** (Counterargument). An argument  $\langle \mathcal{A}_1, h_1 \rangle$  is a *counterargument* to an argument  $\langle \mathcal{A}_2, h_2 \rangle$  at a literal  $h$ , iff there exists a subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$ , such that  $h$  and  $h_1$  disagree.

If  $\langle \mathcal{A}_1, h_1 \rangle$  is a counterargument to  $\langle \mathcal{A}_2, h_2 \rangle$  at a literal  $h$ , then the subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$  is called the *disagreement subargument*. If  $h = h_2$ , then  $\langle \mathcal{A}_1, h_1 \rangle$  is called a *direct attack* on  $\langle \mathcal{A}_2, h_2 \rangle$  and *indirect attack*, otherwise.

**Example 3.** In  $\mathcal{P}$  from Example 1 there is  $\langle \{\neg f(tina) \negleftarrow c(tina)\}, \neg f(tina) \rangle$  a direct attack to  $\langle \{f(tina) \negleftarrow b(tina)\}, f(tina) \rangle$ . Furthermore  $\langle \{\neg f(tina) \negleftarrow c(tina)\}, \neg f(tina) \rangle$  is an indirect attack on  $\langle \{(n(tina) \negleftarrow f(tina)), (f(tina) \negleftarrow b(tina))\}, n(tina) \rangle$  with the disagreement subargument  $\langle \{(f(tina) \negleftarrow b(tina))\}, f(tina) \rangle$ .

A central aspect of defeasible argumentation is a formal comparison criterion among arguments. For some examples of preference criterions see [6]. For the rest of this paper we use an abstract preference criterion  $\succ$  defined as follows.

**Definition 10** (Preference Criterion  $\succ$ ). A preference criterion among arguments is an irreflexive, antisymmetric relation and will be denoted by  $\succ$ . If  $\langle \mathcal{A}_1, h_1 \rangle$  and  $\langle \mathcal{A}_2, h_2 \rangle$  are arguments,  $\langle \mathcal{A}_1, h_1 \rangle$  will be *strictly preferred* over  $\langle \mathcal{A}_2, h_2 \rangle$ , iff  $\langle \mathcal{A}_1, h_1 \rangle \succ \langle \mathcal{A}_2, h_2 \rangle$ .

**Example 4.** A possible preference relation among arguments is *Generalized Specificity* [11]. According to this criterion an argument is preferred to another argument, iff the former one is more *specific* than the latter, i. e. (informally) iff the former one uses more facts or less rules. For example,  $\langle \{c \leftarrow a, b\}, c \rangle$  is more specific than  $\langle \{\neg c \leftarrow a\}, \neg c \rangle$ . For a formal definition see [11,6].

As  $\succ$  is antisymmetric by definition, there is no equipreference among an argument and its counterargument. So we only have to consider the cases, that one argument is better than the other or that two arguments are incomparable.

**Definition 11** (Defeater). An argument  $\langle \mathcal{A}_1, h_1 \rangle$  is a *defeater* of an argument  $\langle \mathcal{A}_2, h_2 \rangle$ , iff there is a subargument  $\langle \mathcal{A}, h \rangle$  of  $\langle \mathcal{A}_2, h_2 \rangle$ , such that  $\langle \mathcal{A}_1, h_1 \rangle$  is a counterargument of  $\langle \mathcal{A}_2, h_2 \rangle$  at literal  $h$  and either  $\langle \mathcal{A}_1, h_1 \rangle \succ \langle \mathcal{A}, h \rangle$  (*proper defeat*) or  $\langle \mathcal{A}_1, h_1 \rangle \not\succ \langle \mathcal{A}, h \rangle$  and  $\langle \mathcal{A}, h \rangle \not\succ \langle \mathcal{A}_1, h_1 \rangle$  (*blocking defeat*).

When considering sequences of arguments, then the definition of defeat is not sufficient to describe a conclusive argumentation line. Defeat only takes an argument and its counterargument into consideration, but disregards preceding arguments. But we expect also properties like *non-circularity* or *concordance* from an argumentation sequence. See [6] for a more detailed description of acceptable argumentation lines.

**Definition 12** (Acceptable Argumentation Line). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* and let  $\Lambda = [\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle]$  be a sequence of arguments.  $\Lambda$  is called *acceptable argumentation line*, iff 1.)  $\Lambda$  is a finite sequence, 2.) every argument  $\langle \mathcal{A}_i, h_i \rangle$  with  $i > 1$  is a defeater of his predecessor  $\langle \mathcal{A}_{i-1}, h_{i-1} \rangle$  and if  $\langle \mathcal{A}_i, h_i \rangle$  is a blocking defeater of  $\langle \mathcal{A}_{i-1}, h_{i-1} \rangle$  and  $\langle \mathcal{A}_{i+1}, h_{i+1} \rangle$  exists, then  $\langle \mathcal{A}_{i+1}, h_{i+1} \rangle$  is a proper defeater of  $\langle \mathcal{A}_i, h_i \rangle$ , 3.)  $\Pi \cup \mathcal{A}_1 \cup \mathcal{A}_3 \cup \dots$  is non-contradictory (*concordance of supporting arguments*), 4.)  $\Pi \cup \mathcal{A}_2 \cup \mathcal{A}_4 \cup \dots$  is non-contradictory (*concordance of interfering arguments*), and 5.) no argument  $\langle \mathcal{A}_k, h_k \rangle$  is a subargument of an argument  $\langle \mathcal{A}_i, h_i \rangle$  with  $i < k$ .

Let  $+$  denote the concatenation of argumentation lines and arguments, e. g.  $[\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle] + \langle \mathcal{B}, h \rangle$  stands for  $[\langle \mathcal{A}_1, h_1 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle, \langle \mathcal{B}, h \rangle]$ .

In DeLP a literal  $h$  is *warranted*, if there exists an argument  $\langle \mathcal{A}, h \rangle$  which is non-defeated in the end. To decide whether  $\langle \mathcal{A}, h \rangle$  is defeated or not, every acceptable argumentation line starting with  $\langle \mathcal{A}, h \rangle$  has to be considered.

**Definition 13** (Dialectical Tree). Let  $\langle \mathcal{A}_0, h_0 \rangle$  be an argument of a *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$ . A *dialectical tree* for  $\langle \mathcal{A}_0, h_0 \rangle$ , denoted  $T_{\langle \mathcal{A}_0, h_0 \rangle}$ , is defined by

1. The root of  $\mathcal{T}$  is  $\langle \mathcal{A}_0, h_0 \rangle$ .
2. Let  $\langle \mathcal{A}_n, h_n \rangle$  be a node in  $\mathcal{T}$  and let  $\Lambda = [\langle \mathcal{A}_0, h_0 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle]$  be the sequence of nodes from the root to  $\langle \mathcal{A}_n, h_n \rangle$ . Let  $\langle \mathcal{B}_1, q_1 \rangle, \dots, \langle \mathcal{B}_k, q_k \rangle$  be the defeaters of  $\langle \mathcal{A}_n, h_n \rangle$ . For every defeater  $\langle \mathcal{B}_i, q_i \rangle$  with  $1 \leq i \leq k$ , such that the argumentation line  $\Lambda' = [\langle \mathcal{A}_0, h_0 \rangle, \dots, \langle \mathcal{A}_n, h_n \rangle, \langle \mathcal{B}_i, q_i \rangle]$  is acceptable, the node  $\langle \mathcal{A}_n, h_n \rangle$  has a child  $\langle \mathcal{B}_i, q_i \rangle$ . If there is no such  $\langle \mathcal{B}_i, q_i \rangle$ , the node  $\langle \mathcal{A}_n, h_n \rangle$  is a leaf.

In order to decide whether the argument at the root of a given dialectical tree is defeated or not, it is necessary to perform a *bottom-up*-analysis of the tree. There every leaf of the tree is marked “undefeated” and every inner node is marked “defeated”, if it has at least one child node marked “undefeated”. Otherwise it is marked “undefeated”. Let  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}^*$  denote the marked dialectical tree of  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}$ .

**Definition 14** (Warrant). A literal  $h \in \mathfrak{L}$  is *warranted*, iff there exists an argument  $\langle \mathcal{A}, h \rangle$  for  $h$ , such that the root of the marked dialectical tree  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}^*$  is marked “undefeated”. Then  $\langle \mathcal{A}, h \rangle$  is a *warrant* for  $h$ .

The notion of warrant is the topic of the next section.

#### 4. Some interesting properties of warrant

The warrant procedure of DeLP is a way to compute the strongest beliefs of an agent. Thus the set of warranted literals (including all facts as they are trivially warranted using the empty argument) can be characterized as a belief set. One important property of belief sets is consistency. In this section we investigate the relationships between warranted literals and especially the consistency of the set of warranted literals.

If a literal  $h$  is warranted and an argument  $\langle \mathcal{A}, h \rangle$  is a warrant for  $h$ , then  $\langle \mathcal{A}, h \rangle$  is considered a “good” argument for  $h$ . But the quality of  $\langle \mathcal{A}, h \rangle$  depends on its position in argumentation lines. If  $\langle \mathcal{A}, h \rangle$  is at the beginning of an argumentation line, then it will be undefeated, as it is a warrant. It is also a “good” argument for  $h$ , if it is at second position in an argumentation line, as the following proposition shows.

**Proposition 1.** *If an argument  $\langle \mathcal{A}, h \rangle$  is undefeated in the dialectical tree  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}$ , then it is undefeated in every dialectical tree  $\mathcal{T}_{\langle \mathcal{A}', h' \rangle}$ , where  $\langle \mathcal{A}, h \rangle$  is a child of  $\langle \mathcal{A}', h' \rangle$ .*

But Proposition 1 can not be generalized to “*If an argument  $\langle \mathcal{A}, h \rangle$  is undefeated in the dialectical tree  $\mathcal{T}_{\langle \mathcal{A}, h \rangle}$ , then it is undefeated in every dialectical tree*”, as the following example shows:

**Example 5.** Consider the following *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  with  $\Pi = \{a_1, a_2, a_3\}$  and  $\Delta = \{(c \prec b), (\neg c \prec \neg d), (\neg d \prec a_1), (d \prec a_1, b), (b \prec a_1, a_3), (b \prec a_2), (\neg b \prec a_3)\}$ . Let *Generalized Specificity* [11] be the preference relation among arguments. The dialectical tree  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$  for the argument  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$  with  $\mathcal{A} = \{(d \prec a_1, b), (b \prec a_2)\}$  consists only of one argumentation line  $[\langle \mathcal{A}, d \rangle, \langle \{\neg b \prec a_3\}, \neg b \rangle, \langle \{(b \prec a_1, a_3)\}, b \rangle]$ .

Observe that the argument  $\langle \{(\neg d \leftarrow a_1)\}, \neg d \rangle$  is not an attack on  $\langle \mathcal{A}, d \rangle$  in  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$ , because  $\langle \mathcal{A}, d \rangle$  is strictly more specific. Thus the argument  $\langle \mathcal{A}, d \rangle$  is undefeated in  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$ . Let  $\mathcal{T}_{\langle \mathcal{B}, c \rangle}$  be the dialectical tree for the argument  $\langle \mathcal{B}, c \rangle$  with  $\mathcal{B} = \{(c \leftarrow b), (b \leftarrow a_1, a_3)\}$ . In  $\mathcal{T}_{\langle \mathcal{B}, c \rangle}$  there is the (incomplete) argumentation line  $\Lambda' = [\langle \mathcal{B}, c \rangle, \langle \{(\neg c \leftarrow \neg d), (\neg d \leftarrow a_1)\}, \neg c \rangle, \langle \mathcal{A}, d \rangle]$ . As in  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$  the argument  $\langle \mathcal{A}, d \rangle$  has exactly one attack in  $\mathcal{T}_{\langle \mathcal{B}, c \rangle}$  after  $\Lambda'$ , namely  $\langle \{(\neg b \leftarrow a_3)\}, \neg b \rangle$ . But different from the situation in  $\mathcal{T}_{\langle \mathcal{A}, d \rangle}$  the argumentation line  $\Lambda' + \langle \{(\neg b \leftarrow a_3)\}, \neg b \rangle$  cannot be extended by the argument  $\langle \{(b \leftarrow a_1, a_3)\}, b \rangle$  as  $\langle \{(b \leftarrow a_1, a_3)\}, b \rangle$  is a subargument of  $\langle \mathcal{B}, c \rangle$  and thus violates the properties of acceptable argumentation lines. Thus  $\langle \mathcal{A}, d \rangle$  is defeated in  $\mathcal{T}_{\langle \mathcal{B}, c \rangle}$ .

Proposition 1 implies an interesting relationship between warranted literals: if an argument  $\langle \mathcal{A}, h \rangle$  is a warrant, every argument  $\langle \mathcal{A}', h' \rangle$  such that  $\langle \mathcal{A}, h \rangle$  is an attack on  $\langle \mathcal{A}', h' \rangle$ , cannot be a warrant. Furthermore due to the definition of warrant, no two warranted literals can disagree.

**Proposition 2.** *Let  $\mathcal{P}$  be a de.l.p.. If  $h$  and  $h'$  are warranted literals in  $\mathcal{P}$ , then  $h$  and  $h'$  cannot disagree.*

Although warranted literals cannot pairwise disagree, the set of all warranted literals might be inconsistent with the strict knowledge as the following example shows:

**Example 6.** Consider the de.l.p.  $\mathcal{P} = (\Pi, \Delta)$  with  $\Pi = \{a, (h \leftarrow c, d), (\neg h \leftarrow e, f)\}$  and  $\Delta = \{(c \leftarrow a), (d \leftarrow a), (e \leftarrow a), (f \leftarrow a)\}$ . In  $\mathcal{P}$  the literals  $c, d, e, f$  are warranted, because for every  $\phi \in \{c, d, e, f\}$  there is the argument  $\langle \{\phi \leftarrow a\}, \phi \rangle$ , which has no counterarguments. But  $\Pi \cup \{c, d, e, f\}$  is inconsistent, as there are derivations for  $h$  and  $\neg h$ . However all pairs and even all triples of  $\{c, d, e, f\}$  are consistent with  $\Pi$  (e. g.  $\Pi \cup \{c, d\} \not\models \perp$ ), as there cannot be derivations for  $h$  and  $\neg h$  from them.

As we want to translate the notion of warrant into the terms of answer set semantics, this property of warranted literals will become a problem, as the literals in an answer set are (jointly) consistent. Because this form of disagreement is not captured in the terms of DeLP we formalize it here as *joint disagreement*.

**Definition 15** (Joint disagreement). Let  $\mathcal{P} = (\Delta, \Pi)$  be a de.l.p. and let  $h_1, \dots, h_n$  be some literals. If  $\{h_1, \dots, h_n\} \cup \Pi \not\sim \perp$ , then  $h_1, \dots, h_n$  are said to be in *joint disagreement*.

If a set  $W$  of literals is given, one might want to determine the literals of  $W$  that are not in joint disagreement. The most primitive construction of a set of literals, that do not jointly disagree, is set up by an argument.

**Proposition 3.** *Let  $\mathcal{P} = (\Pi, \Delta)$  be a de.l.p., let  $\langle \mathcal{A}, h \rangle$  be an argument such that  $\{h, h_1, \dots, h_n\} = \{\text{head}(\delta) \mid \delta \in \mathcal{A}\}$ . Then  $h, h_1, \dots, h_n$  do not jointly disagree.*

Joint disagreement will play a crucial role when converting a de.l.p. into an answer set program in the next two sections.

When considering the set of all warranted literals, another relationship of interest between literals (more precisely between arguments warranting literals) is the subargument relation.

**Proposition 4.** *Let  $\mathcal{P}$  be a de.l.p. and  $\langle \mathcal{B}, h' \rangle$  an argument. If  $\langle \mathcal{B}, h' \rangle$  is defeated in a dialectical process, i. e.  $\langle \mathcal{B}, h' \rangle$  is marked “defeated” in  $T^*(\mathcal{B}, h')$ , every argument  $\langle \mathcal{A}, h \rangle$ , such that  $\langle \mathcal{B}, h' \rangle$  is a subargument of  $\langle \mathcal{A}, h \rangle$ , is also defeated in a dialectical process.*

Due to contraposition Proposition 4 implies directly the following corollary.

**Corollary 1.** *Let  $\mathcal{P}$  be a de.l.p.. If  $h$  is a warranted literal in  $\mathcal{P}$  and  $\langle \mathcal{A}, h \rangle$  is a warrant for  $h$ , then  $h'$  is warranted in  $\mathcal{P}$  for every subargument  $\langle \mathcal{B}, h' \rangle$  of  $\langle \mathcal{A}, h \rangle$ .*

Current algorithms for computing warrant in DeLP only consider computing warrants for one literal [6,3]. If all warranted literals are to be determined, the above results can prune the set of literals to be considered, when the warrant status for one literal has been shown.

## 5. Converting a defeasible logic program into an answer set program

In this section and the next, we present two different conversion techniques to transform a *de.l.p.* into an answer set program. The approach in this section aims at an intuitively correct way to transform defeasible and strict rules into answer set programming. Since the set of all warranted literals might be in joint disagreement, the activation of a transformed defeasible rule must be prohibited when leading to inconsistency. This leads to the notion of minimal disagreement sets.

**Definition 16** (Minimal disagreement set). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*. A *minimal disagreement set*  $\mathcal{X} \subseteq \mathcal{F}(\mathcal{P})$  is a set of derivable literals such that  $\mathcal{X} \cup \Pi \not\vdash \perp$  and there is no proper subset  $\mathcal{X}'$  of  $\mathcal{X}$  with  $\mathcal{X}' \cup \Pi \not\vdash \perp$ . Let furthermore  $\mathfrak{X}(\mathcal{P})$  be the set of all minimal disagreement sets of  $\mathcal{P}$ .

**Example 7.** Consider the *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  with  $\Pi = \{a, b, (h \leftarrow c, d), (\neg h \leftarrow e)\}$  and  $\Delta = \{(p \leftarrow a), (\neg p \leftarrow b), (c \leftarrow b), (d \leftarrow b), (e \leftarrow a)\}$ . The minimal disagreement sets are  $\{h, \neg h\}$ ,  $\{h, e\}$ ,  $\{c, d, \neg h\}$ ,  $\{c, d, e\}$  and  $\{p, \neg p\}$ .

Now joint disagreement can be subsumed by minimal disagreement sets: some literals  $\{h_1, \dots, h_n\}$  are in joint disagreement, iff there is a minimal disagreement set  $\mathcal{X}$  with  $\mathcal{X} \subseteq \{h_1, \dots, h_n\}$ .

Minimal disagreement sets will constrain the derivation of literals in the translated answer set program. If all but one literal of a minimal disagreement set are in the state under consideration, then the derivation of the last literal should be prohibited, in order to maintain consistency of the resulting answer set.

**Definition 17** (Guard literals, guard rules). Let  $\mathcal{P}$  be a *de.l.p.*. The set of *guard literals*  $GuardLit(\mathcal{P})$  for  $\mathcal{P}$  is defined as  $GuardLit(\mathcal{P}) = \{\alpha_h | h \in \mathcal{F}(\mathcal{P})\}$  with new symbols  $\alpha_h$ . The set of *guard rules*  $GuardRules(\mathcal{P})$  of  $\mathcal{P}$  is defined as  $GuardRules = \{\alpha_h \leftarrow h_1, \dots, h_n | \{h, h_1, \dots, h_n\} \in \mathfrak{X}(\mathcal{P})\}$ .

**Example 8.** We continue Example 7. Here we have  $\{(\alpha_h \leftarrow \neg h), (\alpha_{\neg h} \leftarrow c, d), (\alpha_c \leftarrow d, \neg h), (\alpha_c \leftarrow d, e), (\alpha_d \leftarrow c, e)\} \subseteq \text{GuardRules}(\mathcal{P})$

We are now in the situation to propose our first translation of a *de.l.p.* into an answer set program.

**Definition 18** (*de.l.p.-induced answer set program*). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p..* The  $\mathcal{P}$ -induced answer set program  $\text{ASP}(\mathcal{P})$  is defined as the minimal extended logic program satisfying 1.) for every  $a \in \Pi$ ,  $a \in \text{ASP}(\mathcal{P})$ , 2.) for every  $r : h \leftarrow b_1, \dots, b_n \in \Pi$ ,  $r \in \text{ASP}(\mathcal{P})$ , 3.) for every  $h \leftarrow b_1, \dots, b_n \in \Delta$ ,  $h \leftarrow b_1, \dots, b_n, \text{not } \alpha_h \in \text{ASP}(\mathcal{P})$  and 4.)  $\text{GuardRules}(\mathcal{P}) \subseteq \text{ASP}(\mathcal{P})$ .

This translation converts strict and defeasible rules in an intuitively correct manner in ASP-rules. Strict rules are applied whenever possible and defeasible rules are applied whenever consistency is preserved.

**Example 9.** From the *de.l.p.* of Example 7, the complete  $\mathcal{P}$ -induced answer set program  $\text{ASP}(\mathcal{P})$  arises as  $\text{ASP}(\mathcal{P}) = \{a, b, (h \leftarrow c, d), (\neg h \leftarrow e), (p \leftarrow a, \text{not } \alpha_p), (\neg p \leftarrow b, \text{not } \alpha_{\neg p}), (c \leftarrow b, \text{not } \alpha_c), (d \leftarrow b, \text{not } \alpha_d), (e \leftarrow a, \text{not } \alpha_e)\} \cup \text{GuardRules}(\mathcal{P})$  where some guard rules of  $\mathcal{P}$  are as in Example 8.

We now investigate the relationship between arguments in a *de.l.p.*  $\mathcal{P}$  and the answer sets of the  $\mathcal{P}$ -induced answer set program. Let  $\mathcal{F}_\alpha(\mathcal{P}) = \mathcal{F}(\mathcal{P}) \cup \text{GuardLit}(\mathcal{P})$  denote the set of all derivable literals and their guard literals.

**Proposition 5.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*, let  $\langle \mathcal{A}, h \rangle$  be an argument such that  $\{h, h_1, \dots, h_n\} = \{\text{head}(\delta) \mid \delta \in \mathcal{A}\}$ . Let  $S \subseteq \mathcal{F}_\alpha(\mathcal{P})$  be a maximal subset such that 1.)  $\{h, h_1, \dots, h_n\} \subseteq S$ , 2.) for all  $l \in S \cap \mathcal{F}(\mathcal{P})$ , there is an argument  $\langle \mathcal{B}, l \rangle$  such that  $\{\text{head}(\delta) \mid \delta \in \mathcal{B}\} \subseteq S$ , 3.)  $S$  is consistent, i.e. no subset of  $S$  is an element of  $\mathfrak{X}(\mathcal{P})$  and 4.)  $\alpha_l \in S$  iff there is  $X \in \mathfrak{X}(\mathcal{P})$  such that  $X \setminus \{l\} \subseteq S$ . Then  $S$  is an answer set of  $\text{ASP}(\mathcal{P})$ .

**Theorem 1.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.* and  $\text{ASP}(\mathcal{P})$  the  $\mathcal{P}$ -induced answer set program. If  $h$  warranted in  $\mathcal{P}$  then there exists at least one answer set  $M$  of  $\text{ASP}(\mathcal{P})$  with  $h \in M$ .

But as the set of all warranted literals might be in joint disagreement, there can be in general no answer set  $S$  such that all warranted literals are in  $S$ .

**Example 10.** Consider the *de.l.p.*  $\mathcal{P} = (\Pi, \Delta)$  with  $\Pi = \{a, (h \leftarrow c, d), (\neg h \leftarrow e, f)\}$  and  $\Delta = \{(c \leftarrow a), (d \leftarrow a), (e \leftarrow a), (f \leftarrow a)\}$ . The literals  $c, d, e, f$  are warranted in  $\mathcal{P}$ , see Example 6. The  $\mathcal{P}$ -induced answer set program is given by  $\text{ASP}(\mathcal{P}) = \{a, (h \leftarrow c, d), (\neg h \leftarrow e, f), (c \leftarrow a, \text{not } \alpha_c), (d \leftarrow a, \text{not } \alpha_d), (e \leftarrow a, \text{not } \alpha_e), (f \leftarrow a, \text{not } \alpha_f), (\alpha_h \leftarrow \neg h), (\alpha_h \leftarrow c, d), (\alpha_{\neg h} \leftarrow h), (\alpha_{\neg h} \leftarrow c, d), (\alpha_c \leftarrow d, e, f), (\alpha_c \leftarrow d, \neg h), (\alpha_d \leftarrow c, e, f), (\alpha_d \leftarrow c, \neg h), (\alpha_e \leftarrow c, d, f), (\alpha_e \leftarrow f, h), (\alpha_f \leftarrow c, d, e), (\alpha_f \leftarrow e, h)\}$ . The answer sets of  $\text{ASP}(\mathcal{P})$  (without guard literals) are  $\{c, d, e, h\}, \{c, d, f, h\}, \{f, e, f, \neg h\}$  and  $\{c, e, f, \neg h\}$ . Hence, there is no projected answer set  $S$  with  $c, d, e, f \in S$ .

As strict rules are the cause for minimal disagreement sets with cardinality greater than two, we can sharpen the above results for the special case that there are no strict rules.

**Corollary 2.** *Let  $\mathcal{P} = (\Pi, \Delta)$  be a de.l.p. and  $\text{ASP}(\mathcal{P})$  the  $\mathcal{P}$ -induced answer set program. If  $\Pi$  does not contain any strict rule and  $M$  is the set of all warranted literals of  $\mathcal{P}$  then there exists an answer set  $M'$  of  $\text{ASP}(\mathcal{P})$  with  $M \subseteq M'$ .*

If we want to model warrant in general DeLP as a credulous inference from the induced answer set program, then it would be convenient, if we can determine one specific answer set to infer from (as in Corollary 2). This is the topic of the next section.

## 6. A simplified conversion

In this section we present an alternative conversion method to translate a *de.l.p.* into an answer set program. The method presented here is very trivial, but leads to quite stronger results than the above for a special case of preference relation among arguments and also solves the discrepancy described at the end of the last section for arbitrary preference relations.

In [4] the empty preference relation is used to translate a default logic program into a *de.l.p.*. Then the warrant of a literal is equivalent to the sceptical inference of that literal in the original default logic program. By translating a *de.l.p.* into an answer set program, we present here the other direction of this translation.

**Definition 19** (*de.l.p.*<sup>\*</sup>-induced answer set program). Let  $\mathcal{P} = (\Pi, \Delta)$  be a *de.l.p.*. The  $\mathcal{P}^*$ -induced answer set program  $\text{ASP}^*(\mathcal{P})$  is defined as the minimal extended logic program satisfying 1.) for every  $a \in \Pi$  it is  $a \in \text{ASP}^*(\mathcal{P})$  and 2.) for every (strict or defeasible) rule  $h \leftarrow b_1, \dots, b_n \in \Pi \cup \Delta$  it is  $h \leftarrow b_1, \dots, b_n, \text{not } b'_1, \dots, \text{not } b'_m \in \text{ASP}^*(\mathcal{P})$  where  $\{b'_1, \dots, b'_m\} = \{b | b \text{ and } h \text{ disagree}\}$ .

Note that for this conversion into answer set semantics, only pairwise disagreement relations are taken into account. Moreover, strict and defeasible rules are treated likewise. This seems reasonable as Example 10 shows, that strict rules turn out to be the culprits for undercutting a general correspondence between warrant and sceptical inference.

**Example 11.** From the *de.l.p.* of Example 7, the complete  $\mathcal{P}^*$ -induced answer set program  $\text{ASP}^*(\mathcal{P})$  arises as  $\text{ASP}^*(\mathcal{P}) = \{a, b, (h \leftarrow c, d, \text{not } \neg h, \text{not } e), (\neg h \leftarrow e, \text{not } h), (p \leftarrow a, \text{not } \neg p), (\neg p \leftarrow b, \text{not } p), (c \leftarrow b), (d \leftarrow b), (e \leftarrow a, \text{not } h)\}$ . The resulting answer sets of  $\text{ASP}^*(\mathcal{P})$  are  $\{a, b, c, d, e, \neg h, p\}$ ,  $\{a, b, c, d, e, \neg h, \neg p\}$ ,  $\{a, b, c, d, h, p\}$  and  $\{a, b, c, d, h, \neg p\}$ . If the preference relation is *Generalized Specificity* [11], then the set of warranted literals of  $\mathcal{P}$  is  $\{a, b, c, d\}$ .

As one can see for the special case of a *de.l.p.*  $\mathcal{P}$  with no strict rules, the  $\mathcal{P}^*$ - and the  $\mathcal{P}$ -induced translations collapse (in the sense of semantic equivalence). For general DeLP applying the *de.l.p.*<sup>\*</sup>-induced translation yields the following result for warranted literals:

**Theorem 2.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a de.l.p.. Let furthermore  $\text{ASP}^*(\mathcal{P})$  be the  $\mathcal{P}^*$ -induced answer set program. If  $M$  is the set of all warranted literals of  $\mathcal{P}$ , then there exists an answer set  $M'$  of  $\text{ASP}^*(\mathcal{P})$  with  $M \subseteq M'$ .

This theorem states, that every warranted literal can be inferred credulously from its  $^*$ -induced answer set program and even more, that the set of all warranted literals can be inferred credulously using one common answer set. But the inverted statement “If a literal can be inferred credulously, then it is warranted in the original de.l.p.” is not always true as Example 11 shows, where  $e$  can be inferred credulously, but is not warranted.

We investigate now the implications of the above results for the special case  $\text{DeLP}^\emptyset$  of defeasible logic programs with empty preference relation.

**Proposition 6** (Remark 3.4 in [4]). *In  $\text{DeLP}^\emptyset$ , a literal  $l$  is warranted iff there exists an argument for  $l$  that is not attacked.*

When the preference relation under consideration is empty, then warranted literals can be inferred sceptically from the resulting answer set program.

**Theorem 3.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a de.l.p. with the empty preference relation. Let  $\text{ASP}^*(\mathcal{P})$  the  $\mathcal{P}^*$ -induced answer set program and  $M_1, \dots, M_n$  be the answer sets of  $\text{ASP}^*(\mathcal{P})$ . If  $M$  is the set of all warranted literals of  $\mathcal{P}$  then  $M \subseteq M_1 \cap \dots \cap M_n$ .

Equality of  $M$  with the intersection of all answer sets does not always hold. Consider a de.l.p.  $\mathcal{P} = (\Pi, \Delta)$  is given by  $\Pi = \{q, r, h \leftarrow p, h \leftarrow \neg p\}$  and  $\Delta = \{p \prec q, \neg p \prec r\}$ . The  $\mathcal{P}$ -induced answer set program has two answer sets, each of which contains the literal  $h$ . But  $h$  is not warranted as every argument for  $h$  has a defeater attacking either the subargument for  $p$  or  $\neg p$ .

Theorem 3 can also easily give a result for arbitrary preference relations.

**Corollary 3.** Let  $\mathcal{P} = (\Pi, \Delta)$  be a de.l.p. with an arbitrary preference relation. Let furthermore  $\text{ASP}^*(\mathcal{P})$  be the  $\mathcal{P}^*$ -induced answer set program and  $M_1, \dots, M_n$  be the answer sets of  $\text{ASP}^*(\mathcal{P})$ . If  $M' \subseteq \mathcal{F}(\mathcal{P})$  is the set of all literals that have an argument which is not attacked at all then  $M' \subseteq M_1 \cap \dots \cap M_n$ .

Sceptical ASP-inference does not cover all warranted literals for a de.l.p. with an arbitrary preference relation, but so does credulous inference as was shown with Theorem 2.

## 7. Conclusion and future work

Defeasible logic programming provides a framework for paraconsistent reasoning on the basis of dialectical argumentation. Answer set programming is one of the most popular approaches to default reasoning, which is similar to defeasible reasoning in that both methodologies aim at realizing nonmonotonic inferences. In this paper, we studied transformations of defeasible logic programs into answer set programs in order to make relationships between inference via a dialectical warrant procedure, on the one side, and answer set semantics, on the other side,

explicit. We presented two types of conversions that differ with respect to the treatment of strict rules. We proved that for conversions of both types, warrant implies credulous inference. For conversions of the second type, we obtained the stronger result that all warranted literals of the defeasible logic program are contained in one and the same answer set of the transformed logic program. Moreover, in some cases, we were able to show that warranted literals can be inferred skeptically in the answer set environment. In general, however, conversions of the first type establish a much weaker relationship between defeasible logic programming and answer set programming, as strict rules may lead to conflicting defeasible derivations. Of course, in the case that the defeasible logic program does not contain any strict rules, both conversions coincide.

As part of our ongoing work, we will combine our approach with ideas from [4] to obtain a complete picture of the links between defeasible argumentative reasoning in DeLP and answer set semantics. Furthermore it would be interesting to investigate these links when considering an altered version of DeLP using the techniques described in [2].

**Acknowledgments** The authors thank the reviewers for their helpful comments to improve the original version of this paper.

## References

- [1] Ph. Besnard and A. Hunter. Towards a logic-based theory of argumentation. In *Proc. of the 17th American Nat. Conf. on Artif. Intelligence (AAAI'2000)*, pages 411–416, 2000.
- [2] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5-6):286–310, 2007.
- [3] Carlos I. Chesñevar and Guillermo R. Simari. Towards computational models of natural argument using labelled deductive systems. In *Proc. of the 5th Intl. Workshop on Computational Models of Natural Argument (CMNA 2005)*, 2005.
- [4] Telma Delladio and Guillermo R. Simari. Relating DeLP and default logic. *Inteligencia Artificial, Revista Iberoamericana Inteligencia Artificial*, 35:101–109, 2007.
- [5] Phan Minh Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *AIJ*, 77(2):321–358, 1995.
- [6] A. García and G. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1-2):95–138, 2002.
- [7] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.
- [8] Henry Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic*, volume 4, pages 218–319. Kluwer Academic Publishers, Dordrecht, 2 edition, 2002.
- [9] Iyad Rahwan and Leila Amgoud. An argumentation-based approach for practical reasoning. In Gerhard Weiss and Peter Stone, editors, *5th International Joint Conference on Autonomous Agents and Multi Agent Systems, AAMAS'2006*, pages 347–354, 2006.
- [10] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.
- [11] F. Stolzenburg, A. García, C. Chesñevar, and G. Simari. Computing generalized specificity. *Journal of Non-Classical Logics*, 13(1):87–113, 2003.
- [12] M. Thimm and G. Kern-Isbner. On the relationship of defeasible argumentation and answer set programming (extended version). Technical report, Technische Universität Dortmund, 2008.

# Arguments from Experience: The PADUA Protocol

Maya WARDEH Trevor BENCH-CAPON and Frans COENEN

*Department of Computer Science, University of Liverpool  
Liverpool L69 3BX, UK*

**Abstract.** In this paper we describe PADUA, a protocol designed to enable agents to debate an issue drawing arguments not from a knowledge base of facts, rules and priorities but directly from a dataset of records of instances in the domain. This is particularly suited to applications which have large, possibly noisy, datasets, for which knowledge engineering would be difficult. Direct use of data requires a different style of argument, which has many affinities to case based reasoning. Following motivation and a discussion of the requirement of this form of reasoning, we present the protocol and illustrate its use with a case study. We conclude with a discussion of some significant issues highlighted by our approach.

**Keywords.** Argumentation, Dialogue Games, Classification

## 1. Introduction

One application of argumentation which has received a good deal of attention is as the basis of a dialogue between two participants, in which one participant is trying to persuade another of the truth of some claim. In some variations, both participants are trying to persuade the other, and in others the participants are not so committed to a point of view, so that the dialogue takes on the characteristics of deliberation rather than persuasion. A thorough survey of a number of systems can be found in [18]. In this work Prakken identifies the speech acts typically used in such dialogues:

- *claim P* (assert, statement, ...). The speaker asserts that *P* is the case.
- *why P* (challenge, deny, question, ...). The speaker challenges that *P* is the case and asks for reasons why it would be the case.
- *concede P* (accept, admit, ...). The speaker admits that *P* is the case.
- *retract P* (withdraw, no commitment, ..). The speaker declares that he is not committed (anymore) to *P*. Retractions are real retractions if the speaker is committed to the retracted proposition, otherwise it is a mere declaration of non-commitment (e.g. in reply to a question).
- *P since S* (argue, argument, ...). The speaker provides reasons why *P* is the case. Some protocols do not have this move but require instead that reasons be provided by a *claim P* or *claim S* move in reply to a *why* move (where *S* is a set of propositions). Also, in some systems the reasons provided for *P* can have structure, for example, a proof tree or a deduction.

- *question P* (...). The speaker asks another participant's opinion on whether *P* is the case.

These moves seem to suppose that the participant's knowledge is organized in a certain way: namely as a set of facts and, typically defeasible, rules of the form *fact* → *conclusion*. Thus *why P* seeks the antecedent of a rule with *P* as consequent; *P since S* volunteers the antecedent of some rule for *P*, and the other questions suggest the ability to pose a query to a knowledge base of this sort. Prakken's own instantiation of this framework [16] presupposes that the participants have *belief bases* comprising facts, defeasible rules, and priorities between rules. That the participants are presupposed to be equipped with such belief bases doubtless derives in part from the context in which these approaches have been developed. The original example of the approach was probably Mackenzie[12] who was interested in exploring a particular logical fallacy. The take up in Computer Science has largely been by people working in knowledge based systems and logic programming, so that the form of the belief base is a natural one to assume. The result, however, is that the debate takes place in a context where the participants have knowledge (or at least belief), and the dialogue serves to exchange or pool this knowledge. In this way persuasion takes place in the following ways:

- One participant supplies the other with some fact unknown to that participant, which enables the claim to be deduced;
- One participant supplies the other with some rule unknown to that participant, which enables the claim to be deduced;
- An inconsistency in one participant's belief base is demonstrated, so that a claim, or an objection to a claim is removed.

At least one of these must occur for persuasion to happen, but in a complicated persuasion dialogue all three may be required. This necessitates certain further assumptions about the context: that the beliefs of the participants are individually incomplete or collectively inconsistent. Although the participants have knowledge, it is defective in some way, and corrected or completed through the dialogue.

While persuasion dialogues of this form do take place, others take a different form, involving the sharing not of *knowledge*, but of *experience*. In this situation the participants have not analysed their experiences into rules and rule priorities, but draw directly on past examples to find reasons for coming to a view on some current example. One classic example of such reasoning is found in common law, especially as practiced in the US, where arguments about a case are typically backed by precedents. Even where decisions on past cases are encapsulated in a rule, the *ratio decendi*, the particular facts are still considered and play crucial roles in the argument. In informal everyday argument also the technique is common: "the last time we did this, that happened". Given the prevalence of such arguments, it is worthwhile to address the requirements for such dialogues and how they differ from the traditional persuasion dialogues described in [18].

Quite apart from the widespread use of arguments from experience of this sort there are compelling pragmatic reasons for investigating such arguments. The formation of effective belief bases requires a good deal of, typically expensive and skilled, effort. The so called knowledge engineering bottleneck has bedevilled the practical implementation of knowledge based systems throughout their history. If we see the dialogue system as a way of adding value to existing systems, we will find that there are very few suitable belief bases available. In contrast, there are many large datasets available, with each

record in the dataset representing a particular case, a particular experience. This provides an extensive amount of experience to draw on, if we can find a way of deploying it through argumentation.

In the context of these arguments, typically all of the facts regarding the case under consideration are available at the outset. Thus this source of incompleteness, resolved through belief based persuasion dialogues, is not present. Nor can the rules be incomplete or give rise to inconsistency: there are no rules. In such arguments persuasion occurs not through one participant telling the other something previously unknown, but rather because the experience has been incorrectly or unsafely generalised to the current case, or because - importantly - experience differs from participant to participant, and one participant may have encountered an untypical or over narrow set of examples. For example, generalising on experiences confined to the Northern hemisphere, one might conclude that a bird was white on being told that it was a swan, but should be open to persuasion by another participant with experience of Australia also.

## 2. Arguing From Experience

Having seen a need to model arguments from experience, we now need to consider what speech acts will be typical of such dialogues, and see how they differ from those typical of the belief based persuasion dialogues identified by Prakken. One field in which arguing on the basis of precedent examples is Law. Important work has been carried out by, amongst others Rissland, Ashley and Aleven [4,2]. What has emerged from this work is there are three key types of move:

- Citing a case
- Distinguishing a case
- Providing a Counter Example

We will discuss each of these in turn, anticipating the next section by indicating in brackets the corresponding speech acts in the PADUA protocol.

Citing a case involves identifying a previous case with a particular outcome which has features in common with the case under consideration. Given these things in common, the suggestion is that the outcome should be the same. Applied to argument from experience regarding the classification of an example, the argument is something like: *in my experience, typically things with these features are Cs: this has those features, so it is a C* (propose rule). The features in common are thus presented as reasons for classifying the example as *C*, justified by the experience of previous examples with these features.

Distinguishing is one way of objecting to this, by saying why the example being considered does not conform to this pattern. It often involves pointing to features present in the case which make it atypical, so that the “typical” conclusions do not follow. For example the feature may exhibit an exception: *although typically things with these features are Cs, this is not so when this additional feature is present* (distinguish). As an example, swans are typically white, but this is not so for Australian swans. Another form of distinction is to find a missing feature that suggests that the case is not typical: *while things with these features are typically Cs, Cs with these features normally have some additional feature, but this is not present in the current example* (unwanted consequences). Suppose we were considering a duck billed platypus: while we might classify it as mammal on the

basis of several of its features, we would need to consider the objection that mammals are typically viviparous. A third kind of distinction would be to supply a more typical case: *while many things with these features are Cs, experience would support the classification more strongly if some additional feature were also present* (increase confidence). Thus we can have three types of distinction, with differing forces. The first argues that the current example is an exception to the rule proposed; the second that there are reasons to think the case untypical, and so that it may be an exception to the rule proposed.; the third argues no more than that confidence in the classification would be increased if some additional features were present. In all cases, the appropriate response is to try to refine the proposed set of reasons to meet the objections, for example to accommodate the exception.

The point about confidence is important: arguments from experience are typically associated with some degree of confidence: our experience will suggest that things with certain features are often/ usually/ almost always/ without exception *Cs*. This is also why dialogues to enable experience to be pooled are important: one participant's experience will be based on a different sample from that of another. In extreme cases this may mean that one person has had no exposure to a certain class of exceptions: a person classifying swans with experience only of the Northern hemisphere, needs this to be supplemented with experience of Australian swans. In less extreme cases, it may only be the confidence in the classification that varies.

Counterexamples differ from distinctions in that they do not attempt to cast doubt on the reasons, but rather to suggest that there are better reasons for believing the contrary. The objection here is something like: *while these features do typically suggest that the thing is a C, these other features typically suggest that it is not* (counter rule). Here the response is either to argue about the relative confidence in the competing reasons, or to attempt to distinguish the counter example. Thus a dialogue supporting argument from experience will need to accommodate these moves: in the next section we will describe how they are realized in the PADUA protocol.

Another lesson from work on reasoning with legal precedent is the importance of intermediate concepts e.g. [5]. The point is analogous to the difficulty in classifying examples of *XOR* using a single layer perceptron [14]. No simple classification rule for *XOR* over two variables can be produced using only the truth functions of the inputs. Rather we must produce the intermediate classifications “and” and “or” and then classify in terms of these (“or” and not “and”). So too, with law: some features used in classifying cases are not simple facts of the case, but rather classifications of the applicability of intermediate concepts on the basis of a subset of the facts of the case. Dialogues representing arguments from experience must therefore be able to accommodate a degree of nesting, where first the satisfaction of intermediate concepts is agreed, and then used in the main classification debate.

### 3. The PADUA Protocol

In this section we describe PADUA (*Protocol for Argumentation Dialogue Using Association Rules*) an argumentation protocol designed to enable participants to debate on the basis of their experience. PADUA has as participants agents with distinct datasets of records relating to a classification problem. These agents produce reasons for/against

classifications by mining association rules from their datasets using data mining techniques [1,7,9]. By “association rule” we mean no more than that the antecedent is a set of reasons for believing the consequent. In what follows  $P \rightarrow Q$  should be read as “ $P$  are reasons to believe  $Q$ ”. Six of the dialogue moves in PADUA relate to the argument moves identified above. One represents citing a generalization of experience, three pose the different types of distinction mentioned above, one enables counter examples to be proposed, and one enables a rule to be refined to meet objections.

Formally, PADUA is defined drawing on various elements from the different systems suggested in [3,13,17], as the following tuple:

$$< L_t, L_c, A, DP, \varphi, K, L, E, P, O, S > \quad (1)$$

where:

1.  $L_t$ : The topic language of PADUA dialogue game, containing the following elements:
  - (a)  $I = \{i_1, i_2, \dots, i_n\}$ : the set of items. Each item  $i \in I$  has a set of possible values  $V_i$ .
  - (b)  $D$  = the set of database transactions, each transaction  $T \in D$  is a subset of the items in  $I$  such that  $T \subseteq I$ . A transaction  $T$  satisfies a set of items  $X \subseteq I$  if and only if  $X \subseteq T$ .
  - (c) Association rules written as  $ar(P \rightarrow Q, conf)$ :
    - i.  $P$ : rule’s premises.
    - ii.  $Q$ : rule’s conclusion.
    - iii. Each element  $e \in P \cup Q$  is a tuple  $<\text{name}, \text{value}>$  where name is an item  $i \in I$ , and  $\text{value} \in V_i$  is the value of this item in this association rule.
    - iv.  $P \cap Q = \emptyset$ .
    - v.  $conf$ : rule confidence, which means that  $conf\%$  of the transactions in  $D$  that contains  $P$  contains  $Q$  also (i.e. the conditional probability of  $Q$  given  $P$ ).
2.  $L_c$ : The communication language including:
  - (a) *Speech Acts*  $SA = \{\text{propose rule}, \text{distinguish}, \text{unwanted consequences}, \text{counter rule}, \text{increase confidence}, \text{withdraw unwanted consequences}\}$  where:
    - i. *propose Rule*: stands for citing examples in PADUA system by which a new rule with a confidence higher than a certain threshold is proposed. *counter Rule* is very similar and is used to cite counter examples in the same way.
    - ii. *distinguish*: When a player  $p$  plays a *distinguish* move, it adds some new premise(s) to a previously proposed rule, so that the confidence of the new rule is lower than the confidence of the original rule. *increase Confidence* is very similar, except that it increases the confidence of an original rule.
    - iii. *unwanted Consequences*: Here the player  $p$  suggests that certain consequences (conclusions) of the rule under discussion do not match the studied case. *withdraw Unwanted Consequences*: a player  $p$  plays this move to exclude the unwanted consequences of the rule it previously proposed, while maintaining a certain level of confidence.

- (b) *Moves*: each move  $m \in M$  (set of all moves) is defined as a tuple  $\langle sa, content \rangle$  such that:
    - i.  $sa \in SA$ : is the move speech act (or type).
    - ii.  $content$ : is an association rule matches the speech act of the move (except when  $sa = unwanted\ Consequences$  then  $content = U \subset I$  (the set of unwanted consequences)).
  - (c) *Dialogue Moves*: a dialogue move  $dm \in DM$  (the set of all dialogue moves) is defined as a tuple  $\langle S, H, m, t \rangle$  such that:
    - i.  $S \in Ag$  is the agent that utters the move, given by  $Speaker(dm) = S$
    - ii.  $H \subseteq Ag$  denotes the set of agents to which the move is addressed, given by a function  $Hearer(dm) = H$
    - iii.  $m \in M$  is the move, given by a function  $Move(dm) = m$ .
    - iv.  $t \in DM$  is the target of the move i.e. the move which it replies to, given by a function  $Target(dm) = t$ .  $t = \phi$  if  $M$  does not reply to any other move (initial move).
  - (d) *PADUA Dialogues*: defined as a set of finite dialogues, denoted by  $DM^{<\infty}$  the set of all finite sequences from  $L_c$ . For any dialogue  $d = \{dm_1, \dots, dm_n\}$ , the speech act of the first move ( $dm_1$ ) is a propose rule.
3.  $A = \{a_1, \dots, a_n\}$ : The set of participants (players). Each player in PADUA game is defined as:

$$\forall a \in A \quad a = \langle name_a, I_a, C_a, \Sigma_a \rangle \quad (2)$$

Where:

- (a)  $name_a$ : the player (agent) name.
  - (b)  $I_a$ : the set of items this player can understand (i.e. the items included in the player's database).
  - (c)  $C_a$ : the set of classes this player tries to prove that the discussed cases fall under. Each class  $c \in C_a$  is a tuple  $\langle name, value \rangle$  where  $name$  is an item  $i \in I$ , and  $value \in V_i$  is the value this item the player tries to prove it holds.
  - (d)  $\Sigma_a$ : is a representation of the player's background database enables this player to mine for the suitable association rules as needed.
4. *DP*: Is the *dialogue purpose* of PADUA games, defined as the resolution of conflicting opinions about the classification of an instance  $\varphi \subseteq I$ , for example in the case of two players (the proponent *pro* and the opponent *opp*), the proponent may claim that the case falls under some class  $c_1 \in C_{pro}$ , while the opponent opposes the proponent's claim, and tries to prove that case actually falls under some other class  $c_2 \in C_{opp}$  such that  $c_2 \neq c_1$ .
5.  $\varphi$ : The instance argued about i.e. the dialogue subject.
6.  $K \subseteq L_t$ : The *dialogue context* containing the knowledge that is presupposed and must be respected during a dialogue. The context is assumed consistent and remains the same throughout a dialogue.
7. *L*: The *logic* for  $L_t$ .

8.  $E$ : The *effect rules* for  $L_c$ , specifying for each move  $md < p, S, m, t > \in DM$  its effects on the commitments of the participants. We will not discuss effect rules in detail here as they do not relate directly to the subject of this paper.
9.  $P$ : A protocol for  $L_c$  specifying the legal moves at each stage of a dialogue.  $P$  is defined formally as the function:  $P : M \rightarrow 2^M$ , where  $M$  is the set of dialogue acts (moves). Table 1 lists the possible next moves after each move in PADUA protocol.
  - (a) *Termination Rules*: in this version of PADUA, the dialogue ends when a player fails to play a legal move in its turn, in this case, this particular player loses the game while the other player wins it.
  - (b) *Turn taking Rules*: The current PADUA game applies a simple turn taking rule, in which each player is allowed to play exactly one move (speech act) in its turn. The turn taking in PADUA shifts uniformly to all the agents in the dialogue.
10.  $O$ : The *Outcome rules* of PADUA dialogues define for each dialogue  $d$  and instance  $\varphi$  the winners and losers of  $d$  with respect to instance  $\varphi$ .
11.  $S$ : The *Strategy function*

Move	Label	Next Move	New Rule
1	Propose Rule	3, 2, 4	yes
2	Distinguish	3, 5, 1	yes
3	Unwanted Cons.	6, 1	no
4	Counter Rule	3, 2, 1	nested dialogue
5	Increase Conf.	3, 2, 4	yes
6	Withdraw Unwanted Cons.	3, 2, 4	yes

**Table 1.** The protocol legal moves

### 3.1. Nested Dialogues

PADUA allows for dialogues to be nested so that a number of secondary dialogues take place to solve the disputes over some intermediate classifications, before the main dialogue over the main classification starts. To formalize this concept a Control Layer is implemented into the PADUA system. The aim of this layer is controlling the arrangements of the main and secondary dialogues; this layer also manages the communication among the players of every dialogue, to cover the cases in which some players are engaged only in some dialogues, and not in all of them. This layer has been kept as simple as possible, mainly because PADUA dialogues are basically of a persuasive nature. The formalization of the PADUA control layer is defined in the terms of the following components:

1. *Players*: is the set of players engaged in all the PADUA dialogues controlled by this layer.
2.  $G_s$ : set of PADUA secondary dialogue games. Each  $g_s \in G_s$  is defined as an instance of PADUA framework.
3.  $gm$ : PADUA main dialogue game, defined as an instance of PADUA framework.
4. *start*: a function that begins a certain PADUA dialogue game,  $start(g_s \in G_s)$  begins a secondary dialogue game, while  $start(gm)$  begins the main dialogue.

#### 4. Example

To illustrate experimentally the kinds of dialogues produced by PADUA, we applied PADUA to a fictional welfare benefit scenario, where benefits are payable if certain conditions showing need for support for housing costs are satisfied. This scenario is intended to reflect a fictional benefit Retired Persons Housing Allowance (RPHA), which is payable to a person who is of an age appropriate to retirement, whose housing costs exceed one fifth of their available income, and whose capital is inadequate to meet their housing costs. Such persons should also be resident in this country, or absent only by virtue of “service to the nation”, and should have an established connection with the UK labour force. These conditions need to be interpreted and applied [6]. We use the following desired interpretations:

1. Age condition: “Age appropriate to retirement” is interpreted as pensionable age: 60+ for women and 65+ for men.
2. Income condition: “Available income” is interpreted as net disposable income, rather than gross income, and means that housing costs should exceed one fifth of candidates’ available income to qualify for the benefit.
3. Capital condition: “Capital is inadequate” is interpreted as below the threshold for another benefit.
4. Residence condition: “Resident in this country” is interpreted as having a UK address.
5. Residence exception: “Service to the Nation” is interpreted as a member of the armed forces.
6. Contribution condition: “Established connection with the UK labour force” is interpreted as having paid contributions in 3 of the last 5 years.

These conditions fall under a number of typical conditions’ types: conditions (2 and 3) represent necessary conditions over continuous values while conditions (4 and 5) represent a restriction and an exception to the applicant’s residency, condition (1) deals with variables depending on other variables and condition (6) is designed to test the cases in which it is sufficient for some  $n$  out of  $m$  attributes to be true (or have some predefined values) for the condition to be true

A major problem with benefits such as this is that they are often adjudicated by a number of different offices and exhibit a high error rate due to the misunderstanding of the legalisation. This yields large data sets which contain a significant number of misclassifications, the nature of which varies from office to office. To test how PADUA can cope with this situation artificial RPHA benefits datasets (each comprises of 12,000 records) were generated to mimic different systematic misapplications of the rules, such that one does not consider the exceptions to the residency condition (i.e. only UK residents are considered valid candidates for housing benefits), while another interprets the “established connection with the UK labour force” as having paid contributions in 3 of the last 6 years rather than 5. The purpose of this test was to find out whether the proposed dialogue game helps in correctly classifying examples and henceforth correctly interprets them, even when the two agents are depending on (completely or partially) falsely classified examples, this could facilitate the sharing of best practice between offices. Each dataset was assigned to a PADUA player, corresponding association rules were mined from these sets using a 70% confidence threshold for both players, and PADUA was ap-

plied to different sets of examples each of which focuses on an exception of one of the six conditions mentioned above.

In the example discussed below the applicant is a male aged around 70 years, a UK resident who satisfies all the entitlement conditions except that he had paid contributions to the UK labour force in three out of the last 6 years (namely last year, the year before that and 6 years before), this is the case  $\varphi$  argued about between the game players ( $A = \{\text{proponent}, \text{opponent}\}$ ) ; the datasets we use are the ones described in the last paragraph. The dialogue purpose  $DP$  is to decide whether this applicant is entitled to housing benefit or not, where the proponent says he does not ( $C_{\text{proponent}} = \{(entitles, no)\}$ ) nor while the opponent thinks he does ( $C_{\text{opponent}} = \{(entitles, yes)\}$ ).

The dialogue starts with the proponent proposing the rule (R1: `contr y5= not paid -> entitles= no`) with a confidence= 73.14%, the opponent then tries to distinguish this rule in the light of its own experience. For the opponent the rule (R2: `contr y4= not paid, contr y5= not paid -> entitles= no, capital > 3000`) holds with confidence = 2.34% only. This is true because the opponent uses an incorrect interpretation based on its own data, in which the sixth contributions year is considered. This last move is defeated by the proponent by the unwanted consequences attack (`capital>3000` does not hold). The opponent then proposes a counter rule (R4: `age>=65, residence= UK, gross disposable income <20%, 2500 < capital <3000 -> entitles= yes`) with 77.11% confidence, but the proponent can successfully distinguish this rule by emphasizing the fact that the candidate has not paid the contribution fees in the fifth year. The dialogue then progresses in a similar way with the proponent focusing on the unpaid contributions and the opponent trying to get away from this topic in accordance with their own interpretation. For example the proponent proposes the rules: (R13: `contr y3= not paid, contr y5= not paid -> entitles= no`) (88.77% confidence), (R21: `gender= male, contr y3=not paid, contr y5= not paid -> entitles= no`) (confidence = 89.39%). Finally the proponent puts forward the rule (R23: `contr y3= not paid, contr y4= not paid, contr y5= not paid ->entitles= no`), with a confidence of 89.39%, and this rule successfully exposes the mistake in the case under discussion, as by playing this rule the proponent manages to indicate the three years in which the contributions were not paid. The opponent tries then to distinguish this rule by manipulating its premises so it plays the rule (R24: `gender= male, contr y3= not paid, contr y4= not paid, contr y5= not paid -> entitles= no, contr y2= not paid`) in which the confidence falls to 37.89%, but again the opponent's move is defeated by the unwanted consequences attack (the second year contribution is actually paid). The dialogue ends here as the opponent fails to defeat the rule R23 and the proponent wins the game, and the candidate is classified as not entitled to the housing benefits. This game takes 24 moves.

Unfortunately when  $n$  out of  $m$  attributes are needed to decide whether a condition is satisfied or not, like contribution years in our example it is not always the case that the classification process will run correctly. It is more reliable to allow for an intermediate nested dialogue over the contribution years factor, which gives as a result the status of the contribution condition (true or false) before a main dialogue takes place over the eligibility of the applicant. For example if, we take the case  $\varphi_2$  of a male applicant that satisfies all the conditions except for the contribution condition as he paid only the contribution fees of the third, fourth and the sixth years, and apply the one-dialogue

PADUA to this case between the same proponent and opponent as in the last example, the proponent fails to correctly classify the candidate status even after a very exhaustive 30 step dialogue in which contribution years are considered as independent factors and thus the classification is affected directly, as can be shown by some of the rules played in the dialogue:

R1-proponent-Propose Rule: `contr y5= not paid -> entitles= no`  
`confidence= 73.14.`

R23-proponent-Propose Rule: `gender=male, contr y2= not paid,`  
`contr y5= not paid -> entitles= no`  
`confidence=87.69.`

R29-proponent-Propose Rule: `residence=UK, contr y1= not paid,`  
`contr y2= not paid, contr y5= not paid -> age>=65, entitles= no`  
`confidence=95.31`

R30-opponent-Counter Rule: `age>=65, residence=UK, contr y3= paid,`  
`net disposable income <20%, capital <2500 -> entitles= yes`  
`confidence=96.82%`

The latter rule is the final rule in the dialogue, as the proponent fails to defeat it using any of the valid PADUA attacks. Figure 1 shows how, by applying two dialogues (nested and main) to the same case, the proponent becomes able to win the game; and that by winning the nested dialogue over contribution years first, can then apply the result of that dialogue to the main dialogue.

Nested Dialogue	Main Dialogue
(1) – proponent - Propose Rule <code>{contribution y1= not paid, contribution y5= not paid} -&gt; {contribution = no}</code>  <code>confidence=74.71</code>	(1 ) – proponent - Propose Rule <code>{contribution = no} --&gt; {age&gt;=65, entitles=no}</code> <code>confidence=94.0</code>
(2) – opponent - Distinguish <code>{contribution y1= not paid, contribution y3= paid, contribution y5= not paid} --&gt; {contribution = no}</code> <code>confidence=30.0</code>	(2) – opponent - Distinguish <code>{gender = male, contribution = no} --&gt; {age&gt;=65, entitles=no, 2500&lt;capital &lt;3000}</code> <code>confidence=18.85</code>
(3) – proponent - Increase Confidence <code>{contribution y1= not paid, contribution y2= not paid, contribution y3= paid, contribution y5= not paid, contribution y6= paid} --&gt; {contribution = no}</code>  <code>confidence=100.0</code>	(3) – proponent - Unwanted Consequences <code>{gender = male, contribution = no} --&gt; {age&gt;=65, entitles=no, 2500&lt;capital &lt;3000}</code>  <hr/> <code>proponent wins</code>
(4 ) – opponent - Distinguish <code>{contribution y1= not paid, contribution y2= not paid, contribution y3= paid, contribution y5= not paid, contribution y6= paid} --&gt; {contribution = no}</code> <code>confidence=30.0</code>	
(5) – proponent - Increase Confidence <code>{contribution y1= not paid, contribution y2= not paid, contribution y3= paid, contribution y4= paid, contribution y5= not paid, contribution y6= paid} --&gt; {contribution = no}</code> <code>confidence=100.0</code>	
proponent wins	

**Figure 1** Nested and Main Dialogues

## 5. Discussion

PADUA provides a means for agents to engage in discussion about a classification on the basis of raw data, unmediated by knowledge representation effort, to present this data in the form of rules. PADUA necessarily has significant differences from the existing protocols designed to argue about knowledge represented as rules, and the resulting dialogues have a flavour akin to dialogues related to case based reasoning in law. The protocol is particularly applicable to domains in which there are large volumes of data available, where it would prove unrealistic to craft a knowledge base. PADUA can thus complement rule based protocols, since its performance is actually enhanced by large volumes of data, whereas, for example, the work of [8], which used dialogue to generate a rule based theory, can only be applied to comparatively small datasets. Moreover PADUA is ideal for applications with several distributed datasets generated from different samples, since it can exploit and reconcile any systematic differences in the underlying data available to the dialogue participants. Also the work suggested in [10] to generate defeasible and strict rules using association rule mining techniques is limited to small datasets, other restrictions are forced on the datasets used in this work such as they should have no missing values and that all values are correctly recorded.

As it can be viewed as a dialogue game, there is also the question of what strategies and tactics the participants should adopt. Some preliminary work has been done on this [20], there it was shown that the participants can, for example, be represented as cooperative or adversarial. The reported experiments confirm that different strategies give rise to different flavours of dialogue. Some have the flavour of persuasion dialogues, others of deliberation dialogues, demonstrating how these distinct types of dialogue, identified by Walton and Krabbe [19], can be realised in the same protocol when different strategies are used. Further experiments will explore questions relating to how strategies impact on the quality of decisions and the quality of justifications. In [15] an argumentation framework for learning agent is proposed: this framework is similar to PADUA in taking the experience, in the form of past cases, of agents into consideration and focusing on the argument generation process. Yet, the suggested protocol applies learning algorithms techniques, while PADUA implements simpler association rule mining techniques to produce arguments. Also the protocol in [15] is designed for pairs of agents that collaborate to decide the joint solution of a given problem, while PADUA can be applied in variety of situations including persuasion, deliberation and classification.

An important topic of discussion in recent work on reasoning with cases in law is the notion of intermediate predicates (see [5] and [11]). In [11] the important distinction is made between intermediate predicates which are truth functionally determined by some base level predicates, and those for which there is no simple truth functional relationship. For these latter kind of intermediate predicates, it may be necessary to first agree their application before deciding the main question. This is accommodated in PADUA through the possibility of nested dialogues, and the improvements gained were illustrated by an example in the previous section. While this does require some degree of domain analysis to identify and organize the intermediate predicates, so as to form what is termed in IBP [5] a “logical model” of the domain, this analysis is at a high level and, as in IBP, does not require the consideration of individual cases. Once identified this “logical model” can be used by the control layer of PADUA to set the agenda for the dialogue.

Future work will next focus on a set of empirical experiments using a variety of datasets interpreted using a range of misinterpretations and misinterpretations mixtures

to further examine how PADUA can reconcile them. For example we wish to understand how much noise can be tolerated. We also intend to extend PADUA to more than two players as we expect interesting dialogues to come out of such applications, and this is a typical need in the scenarios to which PADUA was applied. Moreover in situations where cases can be classified into more than two categories, adding more players to the game, so that each possibility can have its own advocate, provides a promising solution to such classification problems.

## References

- [1] R. Agrawal, T. Imielinski, A.N. Swami: Association rules between sets of items in large databases. In: Proc. of ACM SIGMOD Int. Conf. on Management of Data, Washington, (1993), 207-216.
- [2] V. Aleven: Teaching Case Based Argumentation Through an Example and Models. PhD thesis, University of Pittsburgh, Pittsburgh, PA, USA. 1997
- [3] L. Amgoud, S. Belabbès, and H. Prade: A formal general setting for dialogue protocols. In: Proc. of AIMA'06 12th Int. Conf. on Artificial Intelligence: Methodology, Systems, Applications, Varna, Bulgaria, (2006), 13 - 15.
- [4] K. D. Ashley: Modeling Legal Argument. MIT Press, Cambridge, MA, USA, 1990.
- [5] K. D. Ashley and S. Brünighaus: A Predictive Role for Intermediate Legal Concepts. In: Bourcier D. (ed) Proceedings of Jurix 2003, IOS Press: Amsterdam (2003), 153-162.
- [6] T.J.M. Bench-Capon: Knowledge Based Systems Applied To Law: A Framework for Discussion. In: T.J.M. Bench-Capon (ed), Knowledge Based Systems and Legal Applications, Academic Press, (1991), 329-342.
- [7] F. P. Coenen, P. Leng and G. Goultbourne: Tree Structures for Mining Association Rules. In: Journal of Data Mining and Knowledge Discovery, Vol 8, No 1, (2004), 25-51.
- [8] A. Chorley, T. J. M. Bench-Capon: AGATHA: Using heuristic search to automate the construction of case law theories. Artificial Intelligence and Law 13(1), (2005), 9-51.
- [9] G. Goultbourne, F. P. Coenen and P. Leng: Algorithms for Computing Association Rules Using A Partial-Support Tree. In: Proc. of ES99, Springer, London, UK, (1999), 132-147.
- [10] G. Governatori and A. Stranieri. Towards the Application of Association Rules for Defeasible Rules Discovery. In jurix, Amsterdam (2001). 63-75.
- [11] L. Lindahl and J. Odelstad: Normative positions within an algebraic approach to normative systems. In Journal of Applied Logic, 17 2, (2005).
- [12] J. Mackenzie: Question-begging in non-cumulative systems. In: Journal of Philosophical Logic 8, (1979), 117-133.
- [13] P. McBurney and S. Parsons: Games That Agents Play: A Formal Framework for Dialogues between Autonomous Agents. In: Journal of logic, language and information, 11(3), (2002), 315-334.
- [14] M Minsky and S. Papert: Perceptrons: An Introduction to Computational Geometry. Cambridge MA: MIT Press, (1969).
- [15] S. Ontañón and E. Plaza. Arguments and Counterexamples in Case-Based Joint Deliberation. In ArgMAS Hakodate, Japan (2006). 36-53.
- [16] H. Prakken: On dialogue systems with speech acts, arguments, and counterarguments. In: Proc. of the 7th European Workshop on Logic for Artificial Intelligence, no. 1919 in Springer Lecture Notes in AI, Berlin. Springer Verlag, (2000), 224-238.
- [17] H. Prakken: Coherence and flexibility in dialogue games for argumentation. In: Journal of Logic and Computation 15 , (2005), 1009-1040.
- [18] H. Prakken: Formal systems for persuasion dialogue. In: The Knowledge Engineering Review 21, (2006), 163-188.
- [19] D. N. Walton and E. C. W. Krabbe: Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning. SUNY Press, Albany, NY, USA, (1995).
- [20] M. Wardeh, T. J. M. Bench-Capon and F. P. Coenen: PADUA Protocol: Strategies and Tactics. In Proc. ECSQARU , 9th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, LNCS 4724, (2007), 465-476.

# Modelling Judicial Context in Argumentation Frameworks

Adam WYNER<sup>a,1</sup>, Trevor BENCH-CAPON<sup>a</sup>

<sup>a</sup>*Department of Computer Science, University of Liverpool, Liverpool, UK.*

**Abstract.** Much work using argumentation frameworks treats arguments as entirely abstract, related by a uniform attack relation which always succeeds unless the attacker can itself be defeated. However, this does not seem adequate for legal argumentation. Some proposals have suggested regulating attack relations using preferences or values on arguments and which filter the attack relation, so that some attacks fail and so can be removed from the framework. This does not, however, capture a central feature of legal reasoning: how a decision with respect to the same facts and legal reasoning varies as the judicial context varies. Nor does it capture related context dependent features of legal reasoning, such as how an audience can prefer or value an argument, yet be constrained by precedent or authority not to accept it. Nor does it explain how certain types of attack may not be allowed in a particular procedural context. For this reason, evaluation of the status of arguments within a given framework must be allowed to depend not only on the attack relations along with the preference or value of arguments, but also on the nature of the attacks and the context in which they are made. We present a means to represent these features, enabling us to account for a number of factors currently considered to be beyond the remit of formal argumentation frameworks. We give three examples of the use of approach: appealing a case, overruling a precedent, and rehearing of a case as a civil rather than criminal proceeding.

**Keywords.**

Argumentation, Legal reasoning, Precedent, Precedence, Procedure

## Introduction

Since their introduction in [1], abstract Argumentation Frameworks (AF) have provided a fruitful tool for the analysis of the acceptability of arguments in a debate, which is comprised of a set of arguments some of which conflict. Generally, arguments are entirely abstract and related only by a uniform attack relation. This attack relation always succeeds: an argument that is attacked can be accepted only if an argument can be found to defeat its attackers. For some applications, however, such as legal argumentation, which will be the focus of this paper, it is useful to allow attacks to fail. Since a court must reach a decision, it requires a rational basis for deciding, for example, between a pair of mutually attacking arguments. For this reason, AFs have been enriched to allow attacks

---

<sup>1</sup>Corresponding Author: Department of Computer Science, University of Liverpool, L69 3BX, UK. Tel.: +44 (0)151 795 4294; Fax: +44 (0)151 795 4235; E-mail: azwyner@liverpool.ac.uk. During the writing of this paper, the first author was supported by the Estrella Project (The European project for Standardized Transparent Representations in order to Extend Legal Accessibility (Estrella, IST-2004-027655)).

to succeed or fail depending on properties of the arguments involved as in preference-based AF (PAF) of [2] or value-based AF (VAF) of [3]. In effect, the success or failure of the attack is *filtered* by these properties so that unsuccessful attacks may be removed, and the results of standard AFs applied.

While VAFs accommodate reasoned choice based on legal principles or social purposes, there are other aspects of legal argumentation, in particular, the notions of *precedent*, *precedence*, and *procedure* as found in *juridical hierarchies* which are not addressed. Precedent here refers to cases which are decided by a court at one point and are subsequently used to guide a decision in another case. Precedence refers to the hierarchical relationships between courts. Procedure refers to what arguments a court finds *legally* admissible relative to some proof standard. In some contexts, while a court may be sympathetic to an argument, the court cannot accept it because that court is obligated to follow a previous decision (precedent), or a decision made by a superior court (precedence), or an argument may be legally inadmissible relative to the court's proof standard (e.g. civil versus criminal proceedings). The nature of the appeals process means that different courts are able to come to different decisions on the same set of arguments. Given these observations, we can see that the evaluation of the status of arguments within a given framework must be allowed to depend not only on the attack relations, nor only on these together with the intrinsic strength of arguments relative to an audience, but also on the ways in which attacks may succeed or fail relative to the contexts and the relationships among contexts in which the arguments and attacks appear. In this paper we will propose a method for accommodating these features using further extensions to AFs.

A set of cases has previously been represented as an AF in [4] and as a VAF in [5]. A means of rewriting VAFs by adding certain auxiliary arguments so that both the object level arguments and arguments expressing preferences between values are included in the framework given in [6]. In this paper we describe a general method to address the contextual issues relating to legal argumentation across juridical contexts. We introduce and ascribe properties to auxiliary arguments; in addition, we give additional structure to the construction of argument networks; the properties are then used with respect to the structure to defend arguments against attackers which are weaker in the appropriate respect. Once the unsuccessful attacks have been removed, we can reduce the structure to an AF. Thus, while our analysis accounts for additional phenomena and adds additional machinery, it benefits from the theoretical results and algorithms which apply to AFs.

We distinguish our approach, where we examine argumentation *across* juridical contexts, from argumentation *within* a juridical context. For instance, [7] focus on the dialectical, dialogical, and procedural aspects of arguments for or against a particular claim *within one legal context*. They model *dialectical* argumentation in terms of premises, rules, and conclusions along with critical questions. Proof standards and burdens of proof may shift *within the legal context* among the parties and so contribute to determining the outcome of that particular case. In contrast, we take the *outcome* of a dialectical argument *within a juridical context* as *input* to our analysis, where we consider outcomes *as the juridical context changes*. In a sense, rather than legal protagonists arguing a case before one court, in our analysis, the courts *themselves* are the protagonists. Thus, issues such as premises and critical questions are not directly relevant to our analysis. Furthermore, we abstract over a range of complexities of proof standards and burdens of proof in order to focus on the *legal* admissibility of an argument. Like [4], we represent a *body of case law*, *not a particular case*; it is, then, more abstract than [7].

The structure of the paper is as follows. Section 1 contains a discussion of relevant aspects of the (English) legal system. In particular, we describe the appeals process, change of use of precedent, and proof standards. We describe the analysis methodically, developing it relative to relevant examples. Section 2 introduces the auxiliary arguments and their interpretation. Section 3 discusses how precedents are set with respect to values in an structured argument network. Section 4 presents the appeals process as a case moves upwards in the legal hierarchy. In Section 5, we show how we accommodate change in the law relative to social change. Proof standards are discussed in Section 6. We end with Section 7 and observations about opportunities for future work developing our approach.

## 1. Judicial Contexts

In this section we consider the aspects of the English Legal System which we address in this paper. Each illustrates how the juridical context can determine the outcome of a case.

### 1.1. Example 1: Appeals Process

The lowest level of the legal hierarchy is the *Crown Court*, where trials on indictment come before a judge and jury. The evidence, legal arguments, and the decision are given according to the procedures specified for the Crown Court. In particular, the Crown Court is *bound* by precedents decided by courts higher in the legal hierarchy. The decisions on points of law made in a Crown Court are not binding on any higher level, nor are they binding on other judges in another Crown Court, though they are *persuasive*. We may refer to a *ratio decidendi* as the legal principle on which the decision is based.

The difference between *binding* and *persuasive* precedents is important. A binding precedent is a decided case which a given court *must* follow in making a decision on the case before it, though this depends on the similarities between the cases. A persuasive precedent is one which is not binding, but which can be applied should it not conflict with a binding precedent and the court which applies the precedent chooses to do so. For our purposes, we simply assert the status of the precedent. We focus on *binding* precedents.

Cases decided in the Crown Court may be appealed to a higher level *Court of Appeals*. Cases can be reconsidered on matters of evidence or of law; for matters of law, there is a claim that the law has been misapplied, the rule of law which was applied is no longer desirable, or some application of the law was inappropriately missed. In effect, the *ratio decidendi* of the prior decision is somehow faulty.

At appeal, judges do not retry the case, but hear the evidence and arguments. The Court of Appeals can overturn a decision of a Crown Court. While the decisions of a Court of Appeals are binding on Crown Courts, the decisions of a higher court are binding on Courts of Appeals. Moreover, a Court of Appeal is bound by the decision of another Court of Appeal, with a range of exceptions (cf. *Young v Bristol Aeroplane Co Ltd* [1944] KB 718). Typically a case in the Appeal Court is heard by three judges.

A case may be appealed from the Court of Appeal to the highest court – the *House of Lords*. The evidence and arguments are heard again, before five judges, called Law Lords. However, the Law Lords who judge the case are not bound by decisions made

at either of the two lower courts. Following *Practice Statement [1966] 3 All ER 77*, the House of Lords is not even obligated to follow its own previous decisions.

### 1.2. Example 2: Change of Use of Precedent

In general it is considered desirable for decisions made in previous cases to be applied in subsequent cases since this makes for consistency of treatment, a greater certainty as to what the law is, and stability in the system. This is the motivation for the ways in which precedents bind decisions as described above. On occasion, however, social changes may make it desirable that precedents are abandoned. This cannot be done lightly, but it is essential that it be possible if courts are to be able to adapt to changes in society at large. An example is provided by *Miliangos v George Frank (Textiles) Ltd [1976] AC 443*, where the House of Lords overruled its own previous decision concerning *Re United Railways [1961] AC 1007* and in favor of allowing damages to be awarded in a foreign currency. This was in response to a radical change in the exchange rate mechanism that had developed in the interim. Prior to 1966, the House of Lords was bound to follow all its prior decisions under the principle of *stare decisis*; however, following the *Practice Statement [1966] 3 All ER 77*, the House of Lords granted itself the right to depart from its previous decisions where it seems right to do so.

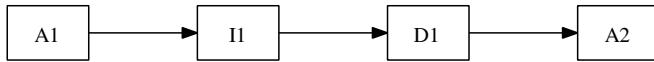
### 1.3. Example 3: Differences in Standards of proof

In criminal proceedings a very high standard of proof, often expressed as "beyond reasonable doubt" is required. Depriving a citizen of his liberty is rightly considered a very serious matter, and a person is presumed innocent until guilt is established. This presumption is very strong: it should be maintained if there are any reasonable grounds for doubt. However, civil proceedings, where the victim seeks compensation, uses a lower standard of proof, termed "balance of probabilities" or "preponderance of evidence". This difference means that on the basis of the same facts, some arguments which were rejected as *legally* inadmissible by the criminal court will be considered and accepted by the civil court. There are a number of examples in fields such as rape, murder, and negligence.

## 2. Representing Legal Context

We take as our starting point an analysis of a *body of case law* which identifies the arguments presented in the cases analysed and the attack relation between them. An analysis of a set of property law cases pertaining to wild animals in terms of a Dungian AF can be found in [4] and in terms of a VAF in [5]. We adapt [5] to accommodate precedent, precedence, and procedure. We consider two fragments of the framework of [5] to highlight components which we use to motivate a basic structure of distinct sorts of arguments in specific attack relations.

First consider two arguments presented in the case *Pierson v Post*. One argument, *PP1*, was that possession of a wild animal required bodily seizure, advanced for the sake of clarity. The other, *PP2*, was that hot pursuit should count as possession, so as to encourage hunting for the sake of economic prosperity. Thus we have a pair of mutually conflicting arguments, each represented with an indexed A node, that can carry different values.



**Figure 1.** An A-Chain

While the representation of [5] records this as the crucial issue on which the decision turns, it does not record what the court decided: rather the VAF gives two sets of acceptable arguments, corresponding to the majority and minority opinions in *Pierson v Post*. In fact, the court decided for *PP1*, giving the majority opinion. This decision needs to be recorded, which we represent with an indexed *D* node that attacks *PP2* and has the interpretation *PP1 defeated PP2*.

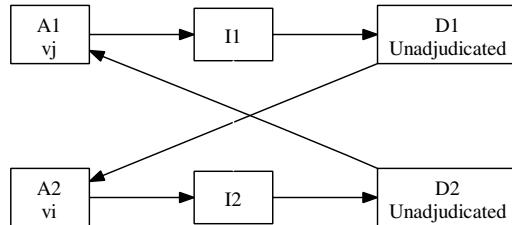
Next consider two arguments presented in *Young v Hitchens*. Argument *YH1* was that the court should find for the defendant because he was in competition with the plaintiff, while argument *YH2* was that the competition was unfair. Both of these arguments were based on the value of promoting economic prosperity. Again the response of the court is not recorded in [5]. In fact the court ruled *YH2* inadmissible, as it was held that deciding what constituted unfair competition was beyond the remit of the court. Thus the argument that *YH2 defeats YH1* is defeated by an argument to the effect that *YH2 is inadmissible*, which we represent with an indexed *I* node. This node is itself attacked by *YH2*.

Putting these two observations together and abstracting over the particulars of the two cases, the attacks between arguments within a case in the framework in [5] effectively comprise a chain of four nodes in an AF, as illustrated in Figure 1.

The arguments taken from the original analysis are indicated with *A1* and *A2*, which we call the *A*-arguments or the *ordinary* arguments. Whereas in [5] the *A*-arguments attacked one another, on our new view, we articulate the attack relations between such arguments so they are *mediated* by intermediary arguments, relating to the possible inadmissibility of *A*-arguments and an assertion about which *A*-argument defeats the other *A*-argument. To represent judicial context, we assume that inadmissibility of an argument is *always* a possibility. We have then *three distinct sorts of arguments*, which we refer to as *A*-arguments, *I*-arguments, and *D*-arguments: the attacking argument is an *A*-argument; an argument that this argument is legally inadmissible is an *I*-argument; an argument that the attacking argument defeats the attacked argument is a *D*-argument; and finally the attacked argument is again an *A*-argument. We refer to the *I* and *D* arguments as the *auxiliary* arguments since they are intended to facilitate reasoning about the ordinary arguments. These arguments can only attack in subsorts of the attack relation: *A*-arguments attack *I*-arguments, the *A*-to-*I* attack; *I*-arguments attack *D*-arguments, the *I*-to-*D* attack; and *D*-arguments attack *A*-arguments (and assuming no argument attacks itself), the *D*-to-*A* attack. We call such a set of arguments as {*A1*, *I1*, *D1*, *A2*} in the specified relationships an *A-chain*, where *A*-arguments appear at the *head* (*A1*) and *tail* (*A2*). We defer discussion of the success of these attacks until additional machinery is introduced.

Assuming the semantics of [1], the preferred extension is {*A1*, *D1*}: *A1* attacks *I1*, where *I1* would only hold were *A1* not to be legally admissible; consequently, the attacker of *D1* is defeated, and *D1* holds; *D1* attacks *A2*.

We enrich the *A*-chain by *labelling* the *A* and *D* arguments. Following [3], we assume that *A*-arguments are labelled with a *value*. We also want to record which of the

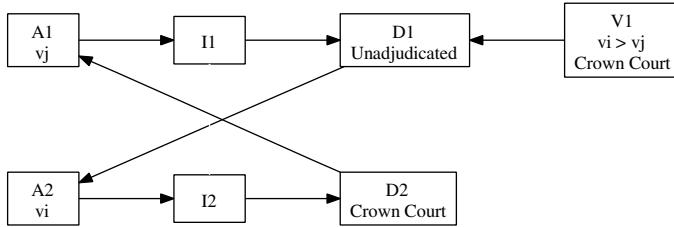
**Figure 2.** An A-Chain with Values and Level of Assembly**Figure 3.** Mutually Attacking Arguments with Values and Level of Assembly

arguments of the form *A defeats B*, the D-arguments, have in fact been endorsed by some assembly (a more general term than court) and the level of assembly which endorsed them. For example, in *Pierson v Post*, *PP2* was endorsed by the lower court, but the decision was overturned by the appeal court which endorsed *PP1*. Thus we label the D-argument which represents *PP1 defeats PP2* with Court of Appeal, and the D-argument which represents *PP2 defeats PP1* with Crown Court. In general, we label the D-arguments with the assembly in which the decision was made. This shall allow us to represent precedent. In some sense, to which we return below, the D arguments indirectly record the *value rankings* of the level of assembly which is used in determining successful attacks. The values of A-arguments and the level of assembly of the D-arguments are *intrinsic* to the arguments. Let us suppose the labels in Figure 2. Here *Unadjudicated* represents that the attack of *A1* on *A2* has not been brought before any judicial assembly.

Let us then assume there is a *disagreement* about which A-arguments win; this will arise where there are *contrary* claims by different parties, one party claiming that *A1* should defeat *A2* and another that *A2* should defeat *A1*. To represent this, we need, then, two A-argument chains. Assuming that our disagreement has not yet been brought before an assembly for adjudication, both D-arguments are labelled with *Unadjudicated*. We then have Figure 3, where we have two A-chains, the heads and tails attacking one another. The A-chain constituted of {*A1,I1,D1,A2*} is called the inverse of {*A2,I2,D2,A1*} and vice versa. This structure yields two preferred extensions {*A1,D1*} and {*A2,D2*}, each representing one claim *winning*. We are now ready to submit the dilemma to an assembly for resolution.

### 3. Precedent

At this point, we want to consider how assemblies adjudicate with respect to A-chains. We first consider where a previously *Unadjudicated* attack is decided by an assembly, establishing a *precedent*. We suppose an attack network such as in Figure 2 is brought before some assembly authorized to adjudicate, meaning that the outcome is disputed and claimed to be otherwise. Recalling that the attacked argument should be able to



**Figure 4.** Structure Decided by an Assembly

withstand the attack of an attacker if it relates to a more highly prized value, we add value rankings. Conceptually, these amount to attacks by a value ranking on the D-argument of an A-chain, for the head defeats the tail only where the D-argument successfully attacks the tail. This represents the relative priority given to values by the adjudicator. We associate these rankings with a level of assembly and assume that every level of assembly has a value ranking, putting aside for the moment just how the rankings and level of assembly are determined. Rather than have such value rankings *external* to the argument network as in [3], we introduce an additional argument sort, the *V-arguments*, which are labelled with the value ranking and the level of assembly which that ranking represents; V-arguments are also auxiliary arguments. In addition, we have two additional subsorts of attack relations where *V-arguments* attack *D-arguments*, V-to-D attacks, and where *V-arguments* attack *V-arguments*, V-to-V attacks, though for brevity these are not used in this paper. V-to-D attacks are *constrained* with *V-to-D Constraint One*: V-to-D attacks can *only* hold where the value ranking of the V argument is contra the values of the head and tail of the A-chain of which the D-argument is a part. Intuitively, the value ranking represented in the V-argument has to represent that argument which would defeat the D-argument given the values of the arguments in the A-chain. This overtly enforces the point that the tail is at least as highly prized as the head. More specifically, the attack of a V on D succeeds where D is an element of an A-chain in which the head of the chain does not have a higher value than the tail of the chain relative to the value ranking of the attacking V; otherwise, the attack fails.

At this point, let us suppose that the attack structure in Figure 2 is brought before and decided by a *Crown Court* which has a value ranking of  $vi > vj$ . We assume that V1 attacks D1 given our constraint on V-to-D attacks. Furthermore, it is useful later to assume *V-to-D Constraint Two*: that for a given D-argument, there can be only one V-to-D attack. As a consequence of this attack, we *construct an inverse A-chain with the D-argument of that A-chain labelled with the level of assembly and add it to the previous structure*. Thus, the level of assembly *stamps* its values on the disagreement. Our system has a series of *steps*, each structure feeding the next. The result is Figure 4. We see how this is iterated below to represent successive steps in the judicial appeal process.

While A1 attacks I1, D1 is attacked by V1, so neither I1 nor D1 hold; therefore A2 can hold. Consequently I2 does not hold (meaning that A2 is legally admissible), which then implies that D2 holds. D2 attacks A1, so it is out. The only preferred extension is {A2, D2, V1}. Thus, the dilemma of choosing between A1 or A2 has been resolved according to the value preference of an assembly. Notice that the conception of successful attack *varies with respect to the arguments in the attack relation and their properties*: a

successful V-to-D attack depends on the values of the head and tail of the A-argument chain relative to the value ranking on the V argument.

In this process, a first *precedent* has been established in the sense that a previously unadjudicated disagreement has been adjudicated, favouring one A-argument over the other.

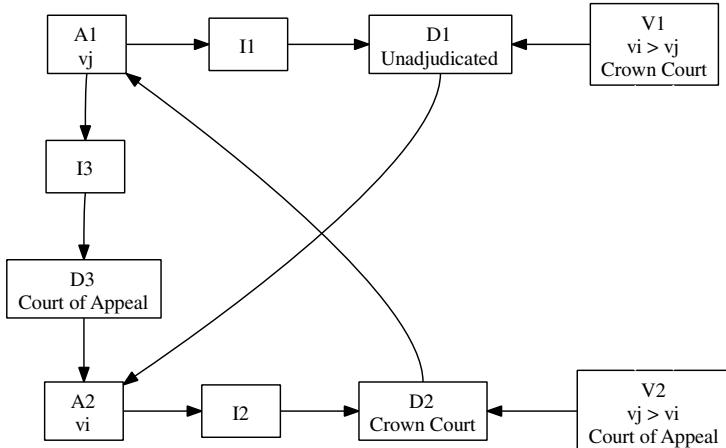
If we consider the structure in Figure 4, we see that we have rewritten an attack between two arguments grounded on values by interposing two auxiliary arguments between them. We have also added a further auxiliary argument, the V argument to attack the D argument. This is the structure proposed for the result of rewriting VAFs in [6] so as to overtly represent attacks on value preferences. Whereas in [6] the auxiliary arguments are abstract, here we have given an interpretation to them, which enables us to ascribe properties to them and handle issues of judicial context.

#### 4. Appeals and Precedence

We now have the essential structure we need to accommodate iterations of judicial consideration, which reflect the appeals process and *precedence* of one assembly over another in the judicial hierarchy. We can now take the decision represented in Figure 4 and suppose that it is appealed to the next level of the judicial hierarchy. We do this by relativising the attacks on D arguments with respect to levels of assembly. An earlier outcome (i.e. preferred extension) can be changed relative to the values or procedures of a higher court. For example, in *Pierson v Post*, *PP2* was endorsed by the lower court, but the decision was overturned by the appeal court which endorsed *PP1*. Thus we label the D-argument which represents *PP1* defeats *PP2* with Court of Appeal, and the D-argument which represents *PP2* defeats *PP1* with Crown Court. To represent the *legal hierarchy* which effects a *change in the outcome of the disagreement* relative to *fixed values*, we *relativise* the V-to-D attacks with *V-to-D Constraint Three*: a V-to-D attack can only appear where the *level of the assembly* on the V-argument and D-argument *abide by the legal hierarchy*. If the legal system requires a strict ordering, then the assembly label on a V-argument must be strictly higher than the assembly label on the attacked D-argument; if the legal hierarchy allows them to be equal, then the assembly labels on V and D can be the same.

Let us iterate the process started in Figure 4. Doing so allows us to preserve the information about the labels on the decisions as well as represent the hierarchical structure of the review process. Given our three constraints on V-to-D attacks, *D2* can only be attacked by a court at the same or higher level and having a value ranking in opposition to that of the A-chain. Assuming a Court of Appeal with value ranking  $vj > vi$  and our construction of an inverse A-chain, we have the three A-chains in Figure 5. The result is that the *V2* attack on *D2* means that *A1* is *reinstated*. While *D1*, which otherwise would have defeated *A2*, is still eliminated by *V1*, we have a new attack of *D3* on *A2*, which is associated with the *Court of Appeal*. The preferred set of arguments is  $\{A1, D3, V1, V2\}$ . This says that while the Crown Court had one preference, the Court of Appeals had the opposite preference, and that the preference of the Court of Appeals took precedence.

It would now be possible to appeal this decision further to the House of Lords, who would be free to endorse the value preference of either of the Lower Courts and so uphold or overturn the decision. We consider this in the next section.



**Figure 5.** Defeat and Value Arguments Labelled with Courts

## 5. Change in Law

The structure in Figure 5 represents a case which has been resolved. Let us elaborate on the interpretation of the V-to-D attack. If the D argument represents information about the level of the court at which it was decided that the head defeats the tail, then a V argument, which is an attack on this decision, represents the values of the assembly *against* this decision. In effect, a V-to-D attack represents the *appeals* process in which the assembly attempts to impose its values with respect to the previous decision of the A-chain to bring about a different result. Our *V-to-D Constraint Three* captures this since only higher level assemblies can attack decisions made by lower level assemblies: Crown Courts only attack unadjudicated arguments, and V1 has imposed its values on a previously unadjudicated case by attacking D1; the Court of Appeals uses its values and ranking relative to the legal hierarchy to attack D2, in effect, overturning the decision of the Crown Court.

The decision now stands as a precedent which must be followed in subsequent similar cases. Whatever the value preference endorsed by future Crown and Appeal Courts, they are required to accept the ruling expressed in Figure 5 since they do not have the status to allow a V argument to attack D3. Suppose, however, that the inclinations of the Crown Court were the first stirrings of a social change, and in the course of time their preference became accepted throughout society. Now a case turning on A1 and A2 will be represented using the same arguments as Figure 5, but now V2 is endorsed only because precedent requires this.

At this point an appeal to the House of Lords will find that assembly subscribing to the preference  $vi > vj$ , which gives us a V-argument, V3, which is labelled for this preference and *House of Lords*. Following the pattern of other appeals, we would introduce into Figure 5 an additional A-chain  $\{A2, I4, D4, A1\}$ , where D4 is labelled with *House of Lords*. This assembly is permitted to use V3 to attack D3, and so A2 is defended. As A2 defeats I4, D4 stands and defeats A1. Thus, the decision by the *House of Lords* reinstates and affirms the original decision of the Crown Court. This decision will stand

as a precedent for subsequent cases in which a conflict between A1 and A2 is material, ensuring that the law is adapted to the changed social climate.

Note that we do not consider the merits of the arguments; our concern is how they have been received by the various assemblies. Moreover, we only discuss a single conflict: in a body of case law, there may be several related conflicts.

## 6. Procedures and Proof Standards

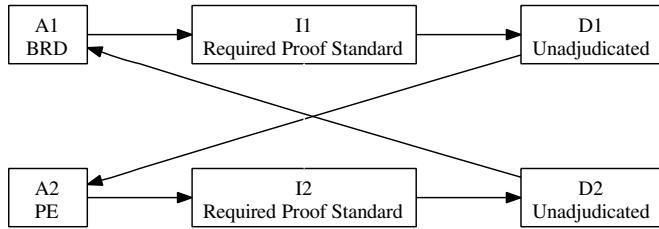
Precedent and Precedence made use of the D and V arguments. In order to reflect the different conditions of *legal* admissibility under different types of procedure, we make use of the I arguments. To simplify the discussion, we omit the V-arguments and values on the A-arguments as well as leave aside issues related to precedent or precedence.

A-arguments which are admitted into the framework will satisfy a particular *proof standard* (PS) with respect to the case under consideration. For our presentation, we abstract over the relationship between proof standards and burdens of proof (see [7]). While [7] discuss four levels of PS arranged in a hierarchy from lower to higher, we need only discuss two.

- Preponderance of Evidence (PE): the strongest defensible pro argument outweighs the strongest defensible con argument, if there is one.
- Beyond Reasonable Doubt (BRD): supported by at least one defensible pro argument, all of its pro arguments are defensible and none of the con arguments are defensible.

Just as we have labelled A-arguments with values, we also label the A-arguments with the PS they satisfy. These are labels which are assigned when the A-arguments appear in the argument network; for brevity, we are *not* representing the pro and con arguments which determine whether a given A-argument satisfies a given PS. In Figure 6, A1 has PS BRD while A2 has PS PE. We refer to the PS on A-arguments as the argument's *evidential status*. However, under different procedures, different PSs are used to determine whether an argument is legally admissible under that procedure: in criminal proceedings BRD is required, whereas PE is enough for civil proceedings. Thus, we are interested to represent the relationship between a *given an evidential status of a particular argument* and the PS in a particular *procedural context*. We therefore represent the procedure under which the case is being considered by labelling the I arguments in the framework with the minimum level of proof required by the procedure *as determined by the assembly in which the case is being considered*. Our assumption is that *when an A-chain is brought before an assembly, just as the D-argument of the new inverse A-chain is labelled for the assembly which makes the decision, the I-argument of that A-chain is labelled for the PS that the assembly uses in making its decision*. In Figure 6, we assume the *Required Proof Standard* is assigned by the procedural context; thus, when the dispute is presented in a criminal court, the required proof standard is set to BRD, while if presented in a civil court, it is PE.

We then assume that the A-to-I attack succeeds only where the evidential status of the A-argument is the same as or higher than the proof standard on the I-argument, otherwise the A-argument is legally inadmissible. Where the I-argument fails, the D-argument can hold (if not attacked by a V-argument). If the A-to-I attack fails, the I-



**Figure 6.** Standards of Proof

argument holds, and it successfully attacks D-argument. In other words, though the head of the chain may have otherwise successfully attacked the tail, the head failed to meet the requisite proof standard, so the attack fails.

For example, in Figure 6, let us assume a criminal case, requiring the PS BRD on I1 and I2. We can start with the A-chain {A1, I1, D1, A2}. Note that we leave D1 to be Unadjudicated, which represents that the assembly *has not made a decision since this is solely a procedural matter*. As the evidential status of A1 is the same as that on I1, I1 is defeated, so D1 survives and successfully attacks A2. In turn, note that in the A-chain {A2, I2, D2, A1}, A2 cannot successfully attack I2, because the evidential status of A2 does not pass the PS on I2. A2 is legally *inadmissible*, and I2 survives, defeating D2. Thus, A1 is not defeated by D2, and {A1, D1, I2} is the preferred extension, whatever the assembly may think about values. Alternatively, consider where the dispute is convened under a civil procedure. The required PS changes to PE, so both I1 and I2 are labelled with PE. Again, if we start with A1, it defeats I1, so again D1 survives and attacks A2. However, now, A2 has a sufficient evidential status to meet the PS, so is legally admissible. Therefore, I2 is defeated, and D2 holds, attacking A1, which is defeated. I1 is reinstated and defeats D1, so A2 survives attack, and {A2, D2, I1} is the preferred extension. By the same token, if we start with A2, {A1, D1, I2} is the preferred extension. We see that as the proof standards change with respect to procedural context, so changes the outcome of the argument attacks. In this latter case, the dilemma would be adjudicated according to the value preference of the assembly relative to the values on the A-arguments.

## 7. Discussion

In this paper we have presented an approach to handling notions of judicial context in argumentation frameworks. Our approach introduces auxiliary arguments with properties in a structured argument network, so that we are able to explicitly express decisions for argument defeat and legal admissibility relative to the assembly. Furthermore, by labelling the decisions and restricting attacks on D-arguments, we are able to account for precedence relations and change of law.

We have illustrated our approach with three examples: appeals and social change which show precedence and precedent, and a change in the nature of proceedings which illustrates variable admissibility. In every case, however, we have restricted ourselves to a single conflict between a pair of arguments. To move to a more complete treatment of all aspects of judicial context we need to explore the following issues.

- Represent a body of case law such as in [5] by merging particular conflicts into cases, and cases into the corpus of decisions.
- Represent majority and minority opinions of assemblies with several judges and different value preferences. We propose to label V arguments with majority and minority preferences, using V-to-V attacks, and specifying that attacks by the V argument with the majority label succeed over the V argument with the minority label. Here we need V-to-V attacks. Similarly, if we wished to allow discussion as to which values should be preferred, the V-to-V attacks would also require auxiliary arguments. Any treatment will need to take account of the Discursive Paradox [8].
- Provide a range of sources of inadmissibility in addition to failure to meet the required PS. For example, evidence derived from illegal search and seizure may be legally inadmissible. This may require us to further articulate the A-to-I attacks with auxiliary arguments.
- Allow additional procedural moves such as the *certiorari* procedure, which allows a previous decision to be quashed and returned to an assembly for reconsideration.
- Incorporate into the analysis *burden of proof* [7], which relates participants in legal contexts to the argument network.

These are just several topics for future work in representing judicial context which have been beyond the reach of representation in AFs. Our approach offers great potential to provide a well-founded representation of arguments in legal case law as well as for other areas where contextual issues are crucial in determining the status of arguments.

## References

- [1] P. M. Dung, “On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games,” *Artificial Intelligence*, vol. 77, no. 2, pp. 321–358, 1995.
- [2] L. Amgoud and C. Cayrol, “On the acceptability of arguments in preference-based argumentation,” in *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, (San Francisco, CA), pp. 1–7, Morgan Kaufmann, 1998.
- [3] T. J. M. Bench-Capon, “Persuasion in practical argument using value-based argumentation frameworks.,” *J. Log. Comput.*, vol. 13, no. 3, pp. 429–448, 2003.
- [4] T. Bench-Capon, “Representation of case law as an argumentation framework.,” in *Legal Knowledge and Information Systems, Proceedings of Jurix 2002* (T. Bench-Capon, A. Dascalopoulou, and R. Winkels, eds.), (Amsterdam), pp. 103–112, IOS Press, 2002.
- [5] T. J. M. Bench-Capon, “Try to see it my way: Modelling persuasion in legal discourse.,” *Artif. Intell. Law*, vol. 11, no. 4, pp. 271–287, 2003.
- [6] S. Modgil and T. Bench-Capon, “Integrating object and meta-level value-based argumentation,” tech. rep., University of Liverpool, 2007.
- [7] T. Gordon, H. Prakken, and D. Walton, “The carneades model of argument and burden of proof,” *Artificial Intelligence*, vol. 171, pp. 875–896, 2007.
- [8] L. A. Kornhauser, “Modelling collegial courts. ii. legal doctrine,” *Journal of Law, Economics and Organization*, vol. 8, pp. 441–470, 1992.

# Author Index

Alsinet, T.	1	Kacprzak, M.	85
Amgoud, L.	13	Kern-Isberner, G.	381, 393
Atkinson, K.	116	Mancini, C.	61
Banihashemi, B.	297	Mann, N.	204
Baroni, P.	25, 37	Martínez, D.C.	216
Bench-Capon, T.J.M.	49, 240, 264, 405, 417	Matt, P.-A.	228
Benn, N.	61	Maudet, N.	285
Besnard, Ph.	v	Modgil, S.	240, 252
Bex, F.	73	Moguillansky, M.O.	336
Buckingham Shum, S.	61, 97	Nawwab, F.S.	264
Budzyńska, K.	85	Norman, T.J.	276
Caminada, M.	109	Oren, N.	276
Cartwright, D.	116	Ouerdane, W.	285
Chesñevar, C.	1	Prakken, H.	73, 252, 324
Coenen, F.	405	Rahwan, I.	297
Dessalles, J.-L.	128	Reed, C.	311, 348
Devereux, J.	311	Rembelski, P.	85
Domingue, J.	61	Riveret, R.	324
Doutre, S.	v, 49	Rotolo, A.	324
Dung, P.M.	134	Rotstein, N.D.	336
Dunne, P.E.	49, 147, 264	Rowe, G.	311, 348
Dupin De Saint Cyr, F.	13	Sartor, G.	324
Efstathiou, V.	159	Sawamura, H.	369
Errecalde, M.L.	171	Simari, G.R.	171, 216, 336
Falappa, M.A.	336	South, M.	360
Ferretti, E.	171	Suzuki, T.	369
Fox, J.	360	Thang, P.M.	134
Gaertner, D.	183	Thimm, M.	381, 393
García, A.J.	171, 216, 336	Toni, F.	134, 183, 228
Giacomin, M.	25, 37	Tsoukias, A.	285
Godó, L.	1	Vreeswijk, G.	360
Hoffmann, M.H.G.	196	Wardeh, M.	405
Hunter, A.	v, 159, 204	Wells, S.	311
		Wyner, A.	417

This page intentionally left blank